EXPOSING AND MITIGATING CROSS-CHANNEL ABUSE THAT EXPLOITS THE CONVERGED COMMUNICATIONS INFRASTRUCTURE

A Thesis
Presented to
The Academic Faculty

by

Bharat Ramakrishnan Srinivasan

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Computer Science, College of Computing

Georgia Institute of Technology
December 2017

EXPOSING AND MITIGATING CROSS-CHANNEL ABUSE THAT EXPLOITS THE CONVERGED COMMUNICATIONS INFRASTRUCTURE

Approved by:

Professor Mustaque Ahamad, Advisor
School of Computer Science
Georgia Institute of Technology

Professor Mostafa Ammar
School of Computer Science
Georgia Institute of Technology

Professor Manos Antonakakis
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Nick Nikiforakis
Department of Computer Science
Stony Brook University

Professor Wenke Lee
School of Computer Science
Georgia Institute of Technology

Date Approved: 27 October 2017

To my parents.

ACKNOWLEDGEMENTS

First and foremost I would like to express my sincere gratitude and appreciation to my advisor Prof. Mustaque Ahamad. He has provided me with unwavering support throughout my Ph.D. journey.

I would also like to thank other members of the dissertation committee, Prof. Manos Antonakakis, Prof. Wenke Lee, Prof. Mostafa Ammar and Prof. Nick Nikiforakis for serving on my committee and for their suggestions in improving this dissertation work. My collaboration with both Manos and Nick significantly improved the quality of work presented in this dissertation.

I am also grateful to the Institute of Security and Privacy (IISP) support staff, Elizabeth Ndongi and Trinh Doan for being ever helpful and taking care of all the administrative paperwork. I would also to like to thank Paul Royal for lending a helping hand in my research especially during the initial years. A special mention goes to Prof. H. Venkateswaran for being a champion of Ph.D. students in the School of Computer Science and providing me with sound advice whenever needed.

I was fortuntate to work with and know Satya Lokam from Microsoft Research, Shachi Sharma and Shalini Kapoor from IBM Research, Sarat Babu and Mahesh U. Patil from Center of Development of Advanced Computing (CDAC). I am thankful to each of them for their mentorship. I would also like to thank Frederick Lee from Twilio Inc. and James Dolph from Salesforce.com for giving me the opportunity to work on interesting security problems faced by industry during my internships in the respective organizations.

My time at Georgia Tech was made enjoyable in large parts due to my labmates, friends and mentors I gained here. A special mention to Sudarsun Kannan, Anand

Louis, Dipanjan Sengupta, Vijay Balasubramanian, Musheer Ahmed, Payas Gupta, Simon Chung, Yacin Nadji, Byoungyoung Lee, Yeongjin Jang, Kadappan Panayappan, Rajan Damle, Ravi Mangal, Shauvik Roy Choudhary, Monjur Alam, Natesh Srinivasan, Sanidhya Kashyap, Pranay Swar, Aditi Ghag, Shardul, Akhil Gupta, Suneet Gupta, Dhanesh Gandhi, Scott Freeman and Leroy Baker.

My sincere gratitude also goes to friends from my alma matter, BITS Pilani and DPS R.K. Puram - Bharat Raju, Dev Basu, Abhinay Pochiraju, Amrit Moharana, Rahul Rai, Deepak Sharma, Kusumakar Althi, Shruti Balaji, Dharashree Panda and Dheeraj Sanka who have all been my well wishers.

Finally, I am indebted to my family for their unconditional love and support. I have greatly admired my father and mother in many ways, for their problem solving skills, hard work, persistence, and indomitable spirit. A special shout out to my dear sister for being younger yet wiser than me and motivating me to purse interests outside of work. I was also blessed by my maternal and paternal grandparents who lived a simple life in spite of the prestigious lineage they inherited from the Chakravarti Rajagopalachari (Rajaji) and Kasturiranga Santhanam families, both towering figures in the Indian Independence movement. I owe many thanks to my aunt, Sita, my uncle, Ramby, Divya, Arjun, Ravi and Shakhila for making me feel at home the moment I stepped into the United States as a student. Last but not the least, I would like to thank my wife for bringing a breath of fresh air in my life and waiting patiently for me to complete this dissertation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Recently we have witnessed rapid consolidation of traditional and emerging communications infrastructures, leading to the convergence of telephony and the Internet. While this convergence has been beneficial in many ways, it has also expanded the arsenal of malicious actors by introducing new attack vectors. Specifically, it offers malicious actors the ability to craft *cross-channel* attacks that combine both telephony and Internet resources to evade existing defenses, abuse the underlying infrastructure and victimize the end-user in ways that have not been adequately explored in the past. In fact, instances of such abuse have attracted the attention of federal law enforcement and consumer protection agencies such as the Federal Trade Commission (FTC) and the Federal Communications Commission (FCC). In response to strong consumer demand for action against *cross-channel* abuse, these agencies have pressed for robust solutions to deal with such abuse.

In this dissertation, we first introduce the notion of cross-channel abuse and place it in the context of traditional notions of Internet and telephony abuse. Then, as a first case in point, using CHURN, a cross-channel messaging attribution system, we present a data-driven longitudinal study of the support infrastructure aiding cross-channel text-messaging abuse which reveals insights into the domain and IP infrastructure used in text-messaging scams, spam and phishing attacks. As a second case in point, using X-TSS, a system developed to track online search-and-ad based cross-channel abuse in technical support scams (TSS), we extend the data-driven approach to study these infamous scams that have plagued consumers and industry brands for over a decade. The lens of a cross-channel view of TSS helps reveal previously underexposed tactics and infrastructure used in these scams. Lastly, based on the

learnings from these two cases, we explore cross-channel intelligence sharing that augments and enhances existing abuse prevention and defense mechanisms on both telephony and Internet channels. By making these contributions, we seek to improve the situational awareness around cross-channel abuse and provide a framework that increases the security and trust of everyday transactions taking place in the converged communications landscape.

CHAPTER I

INTRODUCTION

1.1 Background and Motivation

Over the years there has been a vigorous effort and commendable success in consolidating traditional and emerging communications infrastrucuture, leading to a significant evolution in the entire communications landscape. Starting with the telephone system (PSTNs), then the cellular networks and the Internet (IP network), as new and independent backend infrastructure for large scale and efficient communications were established, the need to make these networks interoperable became paramount. While this interoperability has been enabled by software protocols, hardware devices and provisioned network elements, the benefit is evident with the ubiquitous exchange of text, media and voice based content across these networks. At the end-point, rapid advances in technology now enable simultaneous access to both telephony and Internet services from smart phone devices that people carry with them at all times.

This advancement in infrastructure and end-point technology has enabled linkages between online and telecommunications resources: for instance, online accounts depend on the phone channel for enhancing their security via two-factor authentication. Over the top (OTT) applications, such as `Whatsapp`, that use the Internet to provide connectivity, rely on the telecommunications channel for initial setup and device identification. Authorization of online digital payments in Asia rely on the phone identity (such as mobile device, phone number) [79]. Even online `Bitcoin` wallets are linked to the identities provided by mobile networks [85]. Thus, the online security of users is reliant on the telecommunications channel across industries and this dependency is only likely to increase in the future. We refer to this as the "converged"

communications infrastructure landscape.

The convergence of the Internet with the telecommunications infrastructure has also expanded the arsenal of malicious actors. Specifically, it has offered them the ability to craft cross-channel attacks that leverage both telephony and Internet resources. For instance, text messages delivered over the cellular network via SMS, can contain Internet links that direct unsuspecting users to malicious websites and infrastructure on the Internet controlled by the attackers [93, 12]. VoIP technology, which is an application layer protocol in the Internet protocol suite, can be used to carry out large scale (by means of autodialing services), difficult to trace (by means of CallerID spoofing) and inexpensive fraud campaigns on the phone channel which are hard to detect [68]. As another prominent example, bad actors can leverage the search facility on the Internet channel to lure victims into technical support scams (TSS), via sponsored advertizements (ADs) and search engine optimization (SEO) and then use the phone channel to social engineer the victim [94]. Unless we have visibility into both the telephony and Internet channels such cross-channel attacks make it hard to put the pieces together from a defence perspective.

In other cases, phone numbers have been used to create fraudulent online phone verified email accounts (PVAs), only to be sold in the blackmarket to support underground criminal activities [97]. On the flip side, hackers have been reported to hijack phone numbers to break into email, bank accounts [64] and even Bitcoin Wallets [85]. While these are just examples, such cross-channel attacks often go undetected for long periods of time, thus undermining the levels of trust associated individually with each channel.

In this dissertation, using a data-driven approach, and with automated systems that leverage both unsupervised and supervised machine learning tools, we study and expand the understanding of this new phenomenon of cross-channel abuse that makes use of both the Internet and telephony channels. We then translate our understanding

of cross-channel abuse to explore and design better defence mechanisms, such as identifying infrastructure associated with cross-channel scams, to help mitigate this new class of threats.

1.2 Cross-Channel Abuse in Context

Cybercrime has evolved with technology. In the context of spam, a study published by Levchenko et al. [75] on the end-to-end analysis of the spam value chain, breaks down modern spam into three stages: i) advertizing ii) click support, and iii) realization. Here, advertizement refers to the initial medium used to reach the end users, click support refers to the the websites and domain infrastructure used to further lure the victim into the scam and realization refers to the final monetization portion of the scam. While this classification is based on data mainly from email-based spam containing URLs, it gives us a framework to contextualize the nature of cross-channel abuse.

Table 1 divides the general notion of abuse into three different classes, namely, Internet channel only, telephony channel only and cross-channel, based on which channel is used for each part of the abuse stages. The Internet only abuse that effects online users, sometimes also referred to as online fraud, is the case where all parts of the abuse value chain occur solely over the Internet channel. This is perhaps the most common form of abuse that consists of email based scams, social media scams and plethora of other scams that have persisted over time. Next, we have the telephony only abuse cases, in which all parts of the abuse stages occur solely over the phone. Examples in this category include, the International Revenue Sharing Fraud (IRSF), caller ID spoofing scams and unsolicited robocall scams among others.

Now, we have the relatively new phenomenon of cross-channel abuse, where different stages of the abuse value chain occur over different and independent channels. An

Table 1: Abuse divided into three classes: Internet channel only, telephony channel only and cross-channel.

Advertizement	Support/Monetization	Example(s)	Class
Internet	Internet	Email Scams, Social Media Scams	Internet channel
Telephony	Telephony	IRSF Fraud[1], Caller ID Fraud, Robocalls [68]	Telephony channel
Telephony (Messaging)	Internet	SMS-based Scams with links [93]	Cross-channel
Internet	Telephony (Voice)	Search/AD based TSS [94]	Cross-channel

SMS containing a malicious URL uses the cellular telephony text-messaging infrastructure to advertize/deliver the message and then uses the Internet to support and monetize the scam, like the Internet only case. In the TSS setting, the advertizement of TSS content happens over the Internet channel via sponsored ADs/SEO while the social engineering and monetization happens over the telephony voice channel using the phone number advertized on the TSS websites. More examples of cross-channel abuse have been mentioned previously in Section 1.1. Using the next few sections, we will show that cross-channel abuse presents a new class of threats that requires situational awareness across both channels to develop more effective defence mechanisms that help better protect the operational infrastructure and also the consumers.

1.3 Dissertation Statement

Our hypothesis in this dissertation is that, a data-driven approach can help us understand the evolving nature of cross-channel threats and such understanding can facilitate better use of both Internet and telephony intelligence to mitigate these threats. To support the hypothesis, this dissertation makes the contributions outlined below.

[1]International Revenue Share Fraud (IRSF) is a form of fraud whereby the perpetrator artificially inflates traffic by generating calls to certain portions of international number ranges with no intention to pay for the calls (or paying where there exists some form of arbitrage opportunity), or by stimulating calls by others to the number ranges. The fraudster receives a share of the revenue from termination charges obtained by the number range holder for inbound traffic to the number ranges.

1.4 Contributions

Although both threats and defences in the online and telephony channels have received considerable attention, cross-channel attacks have not been sufficiently explored in the past. We make the following contributions that enhance our understanding of cross-channel attacks.

- We introduce the notion of cross-channel attacks, that leverage both telephony and Internet resources to victimize users. We then place this new class of abuse in the context of traditionally defined Internet and telephony abuse.
- By using a data-driven approach, that leverages several sources of abuse and ground truth data across the telephony and Internet channels, including crowd-sourced telephony intelligence, web threat intelligence on the abusive domain names, web templates and Domain Name System (DNS) intelligence on the abusive IPs, we measure, analyze and understand two distinct cross-channel abuse cases: i) text-messaging abuse, and ii) technical support scams (TSS).
  - To understand cross-channel text-messaging abuse, we design a cross-channel attribution system, `CHURN`, to automate the collection and analysis of SMS-spam abuse containing URLs. The proposed system is able to collect data about large SMS abuse campaigns and analyze their passive DNS records and supporting website properties. It uses a hierarchical clustering technique that employs network level, application level, and popularity-based statistical features to cluster related SMS-spam domain names into campaigns over time. Using CHURN we are able to observe and measure the extent and effectiveness of cross-channel SMS-spam and reveal properties associated with the underlying infrastucture supporting the scam.
  - Technical support scams (TSS) have evolved to use both the Internet and telephony channels to conduct large scale fraud. To understand TSS from

a cross-channel perspective, we develop and deploy a system `X-TSS`, that collects detailed information about search and advertisement tactics used in the latest TSS and then analyzes the underlying infrastructure behind these scams. Using a known corpus of TSS webpages, the system systematically construct queries and uses multiple search engines to find TSS resources such as URIs, redirection chains, domains, and webpages that can be reached either from organic links returned by search engines or advertisements displayed by them. The system also uses network level amplification techniques to discover additional TSS infrastructure that may be difficult to identify via search results and advertisements only. Using the collected data, the system is able to reveal the tactics and infrastructure behind these evolving and sophisticated scams.

- Finally, we show how existing defences such as blacklists on the telephony and Internet channels are insufficient when dealing with cross-channel attacks. We then propose that intelligence generated from systems, such as CHURN and X-TSS, be shared to create a cross-channel security platform that bolsters defences across the two channels from an operational perspective while at the same time increasing the situational awareness around cross-channel abuse.

1.5 Dissertation Overview

This rest of this dissertation is organized as follows. In Chapter 2, we present the related work in the telephony and Internet abuse areas. In the telphony abuse section, we discuss prior work in both messaging and voice abuse. In the Internet abuse section, we discuss prior work relevant to this dissertation, including URL/domain abuse, website abuse detection, passive and active DNS replication and DNS, IP reputation.

In Chapter 3, we present the CHURN system for analyzing cross-channel text-messaging abuse. Using the data gathered by the system for over five years, we conduct a longitudnal study of the infrastructure supporting such abuse and make important observations that can help stem such abuse in the future. Further, we present novel case studies focusing on some of the largest scam campaigns that have used this technique, to defraud consumers, and create new challenges for regulatory and law enforcement authorities.

The chapter on cross-channel text-messaging abuse is followed by Chapter 4, on cross-channel technical support scams (TSS). These scams abuse both the online channel (via search and ads) and the voice channel (via call centers) to victimize end users. First, we present the design of a data-collection and analysis system, called X-TSS that collects online and telephony threat intelligence around cross-channel TSS. Next, using over eight months of threat data, we reveal novel insights into the domain and IP infrastructure supporting cross-channel TSS. We also present case studies that highlight the various techniques used by these scammers to abuse the online channel.

In Chapter 5, we explore how defences on the Internet and telephony channels such as blacklists, can be enhanced by cross-channel intelligence sharing. We provide evidence of the lack of cross-channel intelligence sharing in the existing defence mechanisms and discuss the potential benefits of such sharing including, improving the timeliness of detection of malicious infrastructure associated with cross-channel attacks.

Lastly, in Chapter 6, we conclude this dissertation, with a discussion of both the limitations of the work presented in the dissertation as well as opportunities for future work in this area.

CHAPTER II

RELATED WORK

There has been considerable amount of past research on understanding and combating abuse in the online and phone channels. Rather than provide an exhaustive enumeration of these types of abuse, in this chapter we focus on past work that is most relevant to understanding cross-channel abuse. Since cross-channel attackers leverage both the telephony and Internet channels, we organize the related work into two seperate sections on telephony and Internet abuse respectively. The telephony section is further subdivided into messaging and voice abuse sections. The Internet section meanwhile focuses on aspects of Internet abuse such as URL/domain abuse, passive and active DNS replication, DNS reputation and website abuse detection which are directly relevant to this dissertation. We discuss both past work and put the contributions of this dissertation in the context of these works, highlighting the novelty of these contributions.

2.1 Telephony Abuse

Telephony abuse has been on the rise. In fact, telephony is turning out to be the weakest link in the security chain of trust for cross-channel applications that depend on messaging and voice based communications. This is evident from the increasing number of studies that have highlighted the role of telephony in fraud. Telephony allows an attackers to reach a potential victim either by a message (SMS) or a voice call. A SMS message delivered via the telephony channel can contain a URL that points to malicious content hosted on Internet infrastructure that is controlled by the attacker. Similarly, attackers can abuse Internet-based search content and advertizements to dupe victims into calling a phone number controlled by the attackers. The

attackers can then social engineer the victim into the scam over the voice channel.

2.1.1 Messaging Abuse

SMS-spam is a classic example of messaging abuse that has been on the rise. To collect SMS-spam data, Jiang et al. [71] use the concept of 'grey' phone numbers, which are phone numbers associated with data-only devices such as laptop data cards and electricity meters, as honeypot end points. They then apply statistical models to the collected data to identify the source phone numbers generating spam. Murynets et al. [81] conducted an empirical analysis of SMS-spam collected from fraudulent accounts at a large cellular provider to uncover spamming sources and their strategies. Boggs et al. [60] propose a method to discover emergent malicious campaigns in cellular networks by using graph clustering methods.

Our work on SMS-spam differs from [71], [81] in that ours is one of the first studies to focus exclusively on the characterization of the online network infrastructure supporting SMS-spam. While the analysis in [71], [81] is based on call detail records (CDRs) generated from the telephony channel, and attribution of the source phone numbers generating spam, we explore the cross-channel nature of such abuse by discovering the Internet infrastructure that facilitates this abuse. We use online datasets such as crowd-sourced complaints, passive DNS records and application level information associated with links in SMS-spam messages to discover such infrastructure rather than relying solely on CDRs.

In addition to discovering SMS spam campaigns like in [60], we explore the properties of the infrastructure that supports such campaigns using both passive DNS data and the application level information available from webpages to which users are directed when they click on URLs contained in SMS messages. Our results also show that some of the assumptions made in [60], do not actually hold. For example, [60] assumes that Internet public blacklists can be helpful in detecting

and stopping malicious SMS messages but we show that little overlap exists between domains in SMS messages and these public blacklists.

2.1.2 Voice Abuse

Voice channel abuse is lucrative and effective. Technologies like Voice-over-IP (VoIP) have lowered the cost of making and receiving voice calls while providing anonymity via easy Caller ID manipulation. While this may be desirable in certain use cases, it has also increased the arsenal of voice-based fraudsters.

Vishing (voice phishing) is a common example of voice abuse where a fraudster exploits the phone channel with voice-based interactions to social engineer victims into scams. Maggie et al. [77] conduct one of the first analysis of modern phone frauds relying on vishing. They analyze the content of the conversations in vishing scams, the geography of the target victims, and the role of automation in vishing scams. Robocalls and Caller ID spoofing that have become a major source of voice attacks were addressed by Gupta et al. [68] and Balasubramaniyan et al. [58] in their seminal works respectively. These works contributed to enhancing the defenses on the telephony channel by using an exclusive honeypot for phone calls and by using single-ended audio features such as codec and packet-loss features to detect call provenance, respectively.

Miramirkhani et al. [80] performed the first analysis of technical support scams by focusing on scams delivered via malvertising channels and interacting with scammers over the voice channel to identify their modus operandi. We compare our results with the findings of their study, and show that, while there is some overlap in our findings, our X-TSS discovery system allows us to find scammers that were not discovered by Miramirkhani et al.'s ROBOVIC. We provide a detailed comparison of X-TSS with ROBOVIC in Section 4.5. In recent work, Sahin et al. [86] investigated the effectiveness of chatbots in conversing with phone scammers (thereby limiting the

time that scammers have available for real users).

While most of these prior work focus exclusively on only the voice component of abuse, the exploitation of online services such as search and advertizements by these scams has remained unexplored. Our work on technical support scams explores this aspect, by highlighting the cross-channel nature of certain voice-based scams and investigating the online infrastructure that facilitate the abuse including search, ads, websites and IP infrastructure.

2.2 Internet Abuse related to Cross-Channel Applications

Internet/Online abuse plays a key role in cross-channel scams. The different layers of the Internet protocol suite, the suite of Internet communication protocols and services that enable online interactions, offer an opportunity to collect threat data pertaining to cross-channel abuse cases. Of particular interest are the Layer 3 (IP layer) and Layer 7 (application layer) datasets. There has been prior work in researching Internet/online threats that leverage these datasets. We highlight the prominent works below.

2.2.1 DNS Replication and IP Reputation

Past work has leveraged Layer 3 data to identify threats. The idea of using passive DNS data to capture the relationship between hosts and domain names was motivated by the work of Antonakakis et al. [54] and Bilge et al. [59]. Notos, the system in [54] is able to use properties of how a domain name is used to rank the domain name as potentially malicious or not. Lever et al. [76] analyzed malicious cellular DNS traffic generated by mobile applications to conclude that mobile app-level protection (eg. app-market security) suffices to curtail mobile attacks. Hao et al. [69] proposed SNARE, a spatio-temporal reputation engine for detecting spam messages with very high accuracy and low false positive rates using statistical network-based features to harvest information for spam detection. Prior work has also shown the ineffectiveness

of traditional blacklists in protecting services such as instant messaging (IM) [84], and social media [67, 96].

We borrow ideas from some of the network features presented in [54] and adapt them to the setting of the cross-channel abuse problem. Compared to [76], our work shows that the emergent cross-channel abuse can potentially create an attack vector that bypassses the app-market ecosystem. Our demonstration of the poor blacklist coverage of SMS-spam domains is similar. The significant gap in blacklist coverage and longevity of SMS-spam domains shows the limits of using email and malware abuse intelligence to fight cross-channel abuse. Unfortunately, until blacklist curators adopt systems such as our own, blacklists will also be ineffective against technical support scams.

2.2.2 URL, Domain and Website Abuse

The application layer can be a visible source of abuse. For example, SMS messages with URLs can take unsuspecting victims to attacker controlled domains hosting malicious website content. Prior work has leveraged information available at Layer 7 to identify threats. Thomas et al. [96] designed a real-time system to identify spam URLs submitted to web services such as Twitter. They were able to look at URL-based features such as intial URL, final URL, redirects, frame URL and source URL along with few other features to classify a URL as spam or not. Garera et al. [66] too detect instances of phishing using properties of the URL. Stone-Gross et al. [95] were one of the first to introduce the concept of domain fluxing in which malware uses a domain generation algorithm to generate several domain names, and then communicates with a subset of them. Yadav et. al. [101] proposed a technique to identify botnets by finding randomly generated domain names. Antonakakis et al. [56], use the local DNS query streams to identify new clusters of DGA NXDomains and build models around them. Specificially, they utilize the distribution of the

characters in the domain name. Our contribution is not in detecting the presense or absence of DGAs but in leveraging the existing DGA patterns to help us cluster and label spam campaigns. We do this by embedding domain name features into our clustering and auto-labeling techniques such as using the entropy of the domain name and/or its parts as an indicator of algorithmic structure in domains as well as using the keywords (extracted using an algorithm) present in the domain names to label clusters.

Anderson et al. [53] proposed Spamscatter as the first system to identify and characterize spamming infrastructure by utilizing web sites and images in spam using the image shingling methodology. The X-TSS system too leverages the website content to classify and cluster webpages into different types such as TSS or non-TSS using a bag of words approach or passive vs. aggressive TSS based on features such as popups and alert boxes which is different from the image shingling approach used in Spamscatter.

2.2.3 Search and Advertisement Abuse

Leondatis et al. [74] were one of the first to study the role of search-based attacks in online scams. Specifically, they revealed search-redirection attacks, where miscreants compromise high-ranking websites and dynamically redirect traffic to pharmacies based on the particular search terms issued by the consumer as a means to market and sell illicit prescription drugs. Since TSS is a type of underground ecosystem, we borrowed ideas found in [74], such as, the appropriate setting of *User Agent* and *Referrer* crawler parameters to make requests appear as if they originated from a real user clicking on a search result. Also, search-redirection based drug scams discovered by [74] rely on compromising high-reputation websites while the TSS scams discovered by our system rely on black hat SEO and malicious advertisement tactics.

Chen et al. [62] measure the role of malicious network infrastructure in the online

advertizement bidding process. They find that public blacklists are ineffective in labeling dubious ad publisher domains and propose simple graph analysis to identify such domains. While the study provides useful insights into the ad bidding process itself, it is complementary to our work on how the domains appearing in malicious advertizements, post the ad bidding process, such as technical support scam domains are going undetected by public blacklists. Dave et al. [63] presented a novel system, Viceroi, to catch click spam in search advertisement (Ad) networks. Viceroi operates at the ad network where it has visibility into all ad clicks. While we do not have access to the ad network like Viceroi, our X-TSS system is still able to emulate end user behavior to capture the potentially malicious ads related to TSS.

2.3 Summary

Similar to past work, we collect threat data, extract meaningful features and use them to measure and expose the cross-channel abuse problem. While past work has explored threats in the online or telephony channel, cross-channel threats have not been sufficiently explored in the past. Overall, the individual features used in systems that the dissertation develops may not be novel themselves, our contribution lies in observing that it is most effective to use features from different layers of the network stack in a hierarchical manner so as to capture the fingerprints created by cross-channel threats at different points in the converged communications network.

CHAPTER III

CHURN: CROSS-CHANNEL MESSAGING ATTRIBUTION ENGINE

Recent convergence of telephony with the Internet offers malicious actors the ability to craft cross-channel attacks that leverage both telephony and Internet resources. Bulk messaging services can be used to send unsolicited SMS messages to phone numbers. While the long-term properties of email spam tactics have been extensively studied, such behavior for SMS spam is not well understood. In this paper, we discuss a novel SMS abuse attribution system called CHURN. The proposed system is able to collect data about large SMS abuse campaigns and analyze their passive DNS records and supporting website properties. We used CHURN to systematically conduct attribution around the domain names and IP addresses used in such SMS spam operations over a five year time period. Using CHURN, we were able to make the following observations about SMS spam campaigns: (1) only 1% of SMS abuse domains ever appeared in public domain blacklists and more than 94% of the blacklisted domain names did not appear in such public blacklists for several weeks or even months after they were first reported in abuse complaints, (2) more than 40% of the SMS spam domains were active for over 100 days, and (3) the infrastructure that supports the abuse is surprisingly stable. That is, the same SMS spam domain names were used for several weeks and the IP infrastructure that supports these campaigns can be identified in a few networks and a small number of IPs, for several months of abusive activities. Through this study, we aim to increase the situational awareness around SMS spam abuse, by studying this phenomenon over a period of five years.

3.1 Context and Contributions

While traditional email spamming activities have been extensively studied, long-term properties of SMS spam operations are not well understood by the community. SMS abuse data and long-term network traffic observation of such abuse are necessary to study the behavior of SMS spam operations. By using data that spans a period of close to five years, in this study we aim to present such a long-term analysis of SMS spam abuse. Our hope is that such analysis will provide better understanding of the network properties of SMS spam abuse which can be used to build more effective defenses against it.

We call SMS spam *cross-channel abuse* because it relies on and can be observed in both the telephony and Internet channels. In other words, such attacks involve both a telephony resource (e.g., a phone number) and a traditional Internet resource (i.e., a domain name and/or an IP address). To study cross channel abuse, we explore how SMS spam campaigns utilize the domain name system (DNS) and other Internet infrastructure. We build a SMS spam attribution system called CHURN, which is used to analyze abuse data from a period of five years. CHURN analyzes SMS-spam datasets from two different abuse reporting sources: passive DNS datasets from a large Internet Service Provider (ISP), and application layer web information around these SMS spam campaigns. CHURN's ultimate goal is the attribution of SMS spam campaigns with respect to the domain name infrastructure they employ in their abuse activities.

Our SMS spam attribution analysis reveals that cross channel abuse is highly effective and long lived. We found that the Internet IP infrastructure used by the spammers to support SMS spam campaigns is surprisingly stable. For example, abuse campaigns tend to use a handful of IPs in a few networks over several years to continue their activities. This shows current defenses are either unaware of the abuse infrastructure utilized by SMS spam campaigns or they are not effectively using such

information to combat cross-channel abuse. This chapter demonstrates the value of situational awareness around this problem, which could be used to reduce the potential for social engineering and other attacks facilitated through such cross channel abuse. Summarizing, we make the following contributions in the area of SMS abuse:

- We build and present a cross-channel attribution system to automate the collection and analysis of SMS spam abuse. Our system, namely CHURN, uses a hierarchical clustering technique that employs network level, application level, and popularity-based statistical features to cluster related SMS spam domain names into campaigns over time.

- Using CHURN, we conduct a five year study that yields attribution results for a plethora of real world SMS spam campaigns. We use (1) 8.32 million SMS abuse reports that consist of messages that directed users to scam websites, (2) more than 56 thousand DNS resource records related to the SMS abuse reports since 2011, and (3) more than 67 thousand web pages reflecting the application layers of the SMS spam campaign. Our experiment helps us conclude the following:

  - We show that a mere 1% of SMS abuse domains appear on public Internet domain blacklists. Among the blacklisted domain names, 94% appeared on blacklists weeks or even months after they were first seen in abuse reports.

  - We show that the domains are long lived during the period of abuse with over 40% of the SMS spam domains being active for over 100 days.

  - We dive deep into the three largest and most long-lived case studies of SMS spam campaigns identified by CHURN. We show that (1) spammers were able to operate these campaigns for more than three years, (2) they consistently used a handful of IPs in a few abuse friendly networks, and (3) the average SMS spam domain name lifetime was in the order of two

months, further emphasizing the lack of situational awareness around such cross-channel threats.

3.2 Background

Spammers have been evolving their operations for more than a decade. It comes as no surprise that as Internet defenses are bolstered, the telephony channel has become an attractive spam target. To better understand this, we aim to study the properties of unsolicited bulk SMS messaging (a.k.a. SMS spam) containing URLs with respect to the Internet infrastructure that supports this abuse. Since the attack relies on both telephony and Internet infrastructure (e.g., domains included in SMS spam URLs and associated IPs), we refer to this problem as "cross-channel abuse". In this section, we provide a high-level overview of the ecosystem that facilitates this cross channel abuse, as can be seen in Figure 1.

Delivering SMS Spam at Scale: To successfully "trick" users into scam operations, spammers need a way to reach potential victims. Because phone numbers come from a limited name space with a defined format, they can be auto-generated randomly or picked selectively. Armed with phone numbers, fraudsters can accomplish large scale distribution of SMS spam in several ways.

1. **Disposable SIMs:** Spammers can purchase disposable subscriber identification module (SIM) cards with gateways having slots to hold hundreds of them or use stolen cell phones and USB modems/Aircards [81] as an entry point into the cellular networks. They can then program these devices using off the shelf bulk SMS software or even Arduino [52] micro-controllers to send well crafted bulk SMS spam.

2. **Exploiting Cloud Telephony Services:** Legitimate cloud telephony Infrastructure as a Service (IaaS) providers such as Twilio [48] and Tropo [47], or

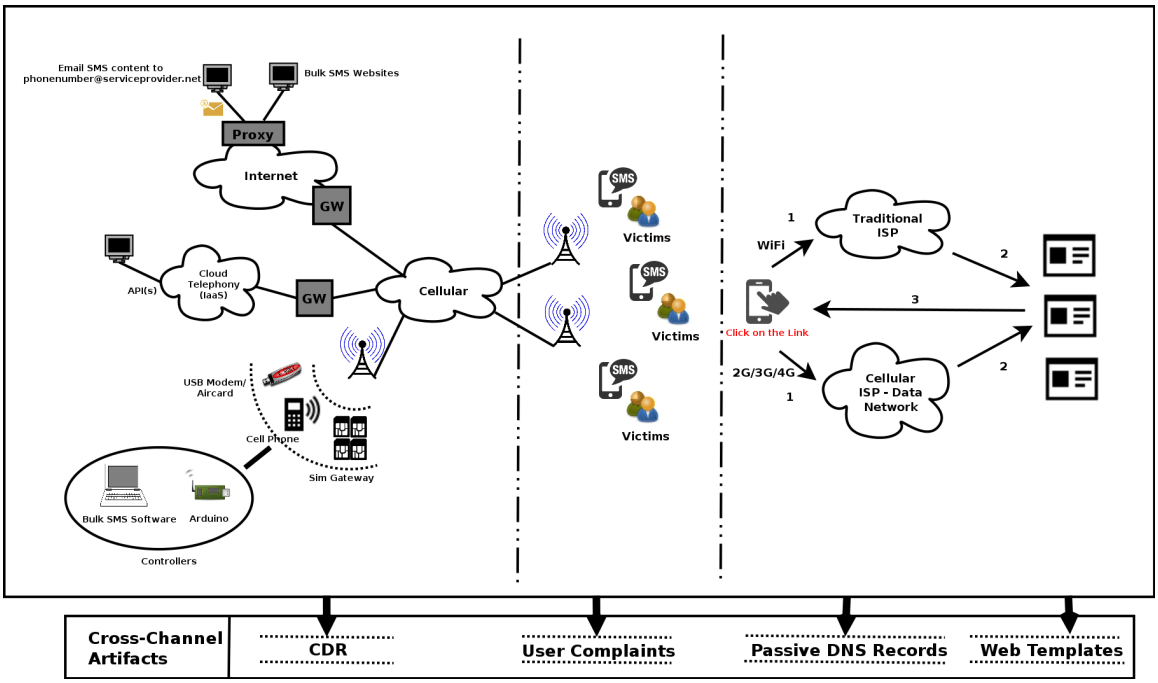


Figure 1: Illustration of the cross-channel abuse ecosystem, delivery mechanisms and potential observation points.

even cellular ISPs [81], can be abused by spammers to deliver bulk SMS messages. This is achieved in one of three ways: (1) creating fraudulent accounts on these platforms, (2) hijacking existing (legitimate) accounts, or (3) exploiting unprotected SMS application programming interfaces (APIs) that allow users to transmit a large volume of SMS messages in an automated fashion[1].

3. **Bulk SMS Services:** Spammers can exploit or collude with existing bulk SMS services to deliver messages. Sometimes, services offered by legitimate service providers enable bridging of the email and SMS mediums by allowing email to be sent as an SMS (or vice versa). This can be abused by spammers.

Monetization: After delivering the spam SMS messages, in order for monetization spammers lure victims into responding to, or interacting with, the message. Specially

[1]Although Twilio and others have a policy against such abuse [51], spammers often find ways to violate it [40].

crafted messages with easy-to-click URLs provide an effective way to automate such response. On smartphone-like devices, victims can simply click these URLs and visit a traditional web site that will lure them into the scam. The key point here is that, while the attack vector clearly started as a telephony based communication (vis-à-vis, the SMS spam), these spammers will often try to social engineer the user into a scam using traditional Internet resources. There are multiple reasons to do this, from minimizing the forensic trail in the telephony network to re-utilizing already provisioned Internet infrastructure for abusive actions. Often the content of such illicit webpages can be tailored to the specific scam.

Observing Cross-Channel Abuse: Cross-channel abuse can be observed in both the telephony and Internet channels. Prior work in combating telephony abuse mainly relied on call detail records (CDRs) to identify and block phone numbers that originate spam SMS messages [81, 71]. Cross-channel abuse also requires traditional Internet resources to direct victims to scam websites. This provides an opportunity to observe such communications by passively monitoring network traffic (i.e., the DNS resolutions). For example, when the recipient of an SMS message clicks an embedded link, it typically initiates a DNS resolution process. The end result of this resolution process is the mapping between the requested domain and the IP address hosting it. The client device typically requests the web page associated with the clicked link from the resolved IP address. The DNS visibility at the ISP (cellular or otherwise) recursive resolver level can serve as a great vantage point to study the SMS spam cross-channel abuse with respect to the Internet channel.

3.3 Cross-Channel Attribution Engine

In this section, we discuss the details of our Cross Channel Attribution Engine called CHURN. The goal of CHURN is to help understand SMS abuse by attributing domain names in SMS-spam campaigns. CHURN achieves this by clustering network

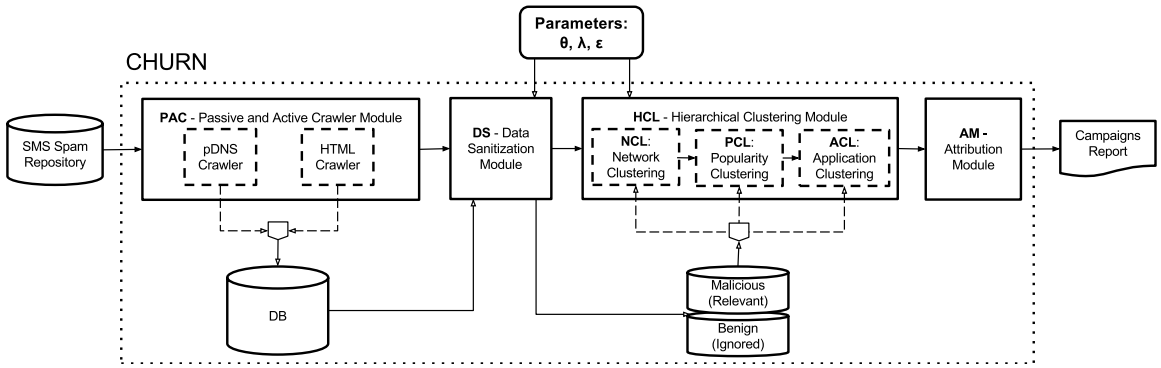


Figure 2: CHURN: The cross-channel messaging attribution engine.

(i.e., domain names and IPs) and application (i.e., HTML content) layer signals that facilitate a given spam campaign. CHURN starts with crowd sourced abuse complaints and produces attributed campaigns with associated network resources. To accomplish this, it performs four tasks serially, as can be seen in Figure 2. Next, we describe in detail each of these four tasks.

3.3.1 Data Collection Module

Our data collection module takes as input external data source(s) of known SMS-spam. In our case, this dataset comes from two sources: (i) SMS-spam complaint reports filed by consumers to the Federal Trade Commission (FTC) [13], which were made available to participants in the Robocall Challenge [17], and (ii) publicly available SMS complaint reports from the online portal SMS watchdog [41][2]. While reports from SMS watchdog were crawled between Jan 2011-Aug 2015, the FTC complaint records were limited to the period Jan 2011-Dec 2012 consisting of reports with anonymized destination numbers. Using SMS messages from user complaints as input, we extract the source (e.g., phone number), timestamp t_d, and URL from each SMS-spam report. Using the URLs, we actively crawl different public and private data sources, which provides information about both the website and the network

[2]smswatchdog.com was down when we last checked as on 02/18/2016 but snapshots of it can be found on the Wayback Machine [25].

hosting infrastructure facilitating the scam.

Passive DNS Dataset (Network Intelligence): Cross channel attacks, like users responding to SMS-spam messages, can be observed in the Internet when the recipient of the message clicks on the URL of a spam message. In this case, a DNS resolution request will be observable at the local recursive DNS servers. This forensic signal cannot be used to estimate the global abuse properties of a particular SMS-spam campaign, as it is non-trivial to obtain global visibility in the DNS recursive plane. However, given a large enough recursive DNS visibility, it could provide forensic evidence and lower bounds on the following three questions: (i) how long was the campaign active, (ii) what was the average lookup volume and a lower bound on the victims that were targeted by each SMS-spam message, and (iii) what was the domain name and IP network infrastructure that supported this cross channel abuse?

By gaining access to a large private passive DNS repository, we were able to collect datasets that could answer these three questions for every domain name contained in our SMS-spam abuse dataset. As we will discuss in subsection 3.3.3, the passive DNS (pDNS) dataset plays an important role in our effort to statistically describe the network properties of SMS-based abuse.

HTML Crawler (Application Intelligence): We implement both an active and a passive method to collect datasets that capture application layer properties of the SMS-spam websites. We download and store the full HTML source from the web page pointed to by each URL seen in SMS-spam reports. In many cases, however, the websites of interest were taken down before we could recover any useful intelligence. For such cases, we relied on the Wayback machine [25].

3.3.2 DS: Data Sanitization Module

The lifecycle of a spam domain involves multiple phases. In the first phase, when the threat is active, the domain will point to IP infrastructure that facilitates the spam operation. Once the spam operation is over, or the domain simply ceases to be used by the spammers, it will enter a phase when it is "parked" or is taken down by network defenders or eventually expires. From the threat analysis and attack attribution point of view, we care to analyze the network infrastructure when the domain is actively used by a spam campaign. The goal of the sanitization module is to weed out the benign infrastructure (in the form of legitimate IP addresses) and HTML sources (related to parked domains) while retaining the network and application information that can be used to analyze the campaigns. Next, we discuss in detail how we can achieve this sanitization of the datasets.

Filtering the pDNS Datasets: Among the domains included in the URLs received in the complaints, we first remove any records containing domains historically appearing in the Alexa [4] top 1 million ranks since 2011. We were able to remove 715 domains using this filter. Next, we use two heuristics to remove DNS information that is related to legitimate IP infrastructure from our datasets. The first heuristic aims to capture the *popularity* of the infrastructure supporting a domain. Parking IP address space is often used to host a relatively large number of domains, at least that is how "domaineers" operate. The number of resource records per IP is a good measure of this as it encapsulates both the diversity in the domains and the popularity in DNS lookup value to domains hosted on certain IPs. The second heuristic aims toward the characterization of the *name server* list supporting a domain. Some name servers (NS) are well known to be associated with parking activities, as they do not try to hide. We create a hard curated list of such name servers using publicly available information and prior work [99].

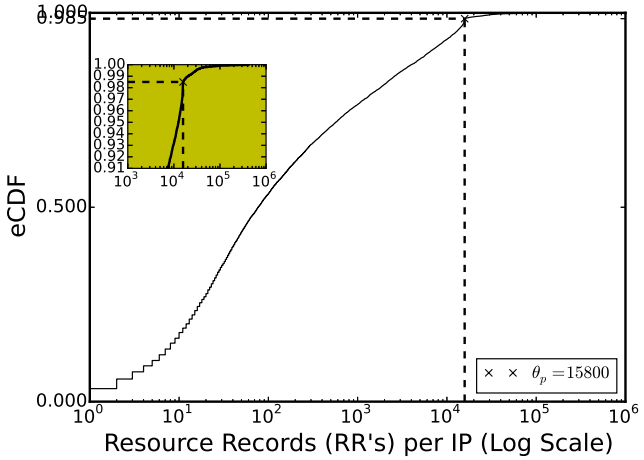


Figure 3: CDF of Resource Records per IP with cut-off threshold θ_p.

More precisely, given a set of pDNS resource records denoted by RR, the sanitization module uses a filter method that uses parking IP threshold θ_p and a name server list, denoted by NS, to create a filtered set, $RR^{\theta_p,NS}$, which consists of all $rr \in RR$ s.t. (i) IP in rr is pointed to by $< \theta_p$ resource records, and (ii) the name server for the domain name d in the $rr \notin \{\text{NS}\}$. Figure 3 shows the cumulative distribution function (eCDF) of the number of resource records hosted per IP in our dataset and the cut-off threshold θ_p. In total we were able to identify $\sim$1% (232 out of 23,269) IPs as parking and ignore records associated with them for the shown value of θ_p.

Filtering Application-level Data: To identify the full HTML sources relating to parked domains, we built a supervised binary classifier to identify if an HTML source file was related to a parked domain or not. To train our classifier, we used 20 features extracted from HTML sources. These features included number of links in the source, number of unique domains in the links, minimum, maximum and average link length, number of external links, ratio of internal to external links, website directory presence, source length, text to html ratio based on the number of characters, presence of Javascript redirect and meta refresh redirection mechanisms, boolean value for if the

Table 2: Confusion matrix for the parking classifier.

	Predicted NP	Predicted P	Total
Actual NP	197	3	200
Actual P	1	199	200
Total	198	202	

meta domain was external, number of frames and iframes and respective number of distinct frame and iframe domains and boolean values to indicate if any of the iframe or frame domains were external. We also counted the number of images present in the HTML source. Intuition behind these features can be found in the work by Vissers et al. [99].

We trained the SVM model [61] using the 10-fold cross validation technique on a set of 200 parking and 200 non-parking feature vectors extracted from webpages in our dataset. With a threshold of 0.5 we were able to achieve a reasonable TPR of 99.5% and FPR of 1.5%. Table 7 shows the confusion matrix using 10-fold cross validation related to this experiment, where NP denotes non-parking webpages and P denotes parking webpages. In total, the classifier was able to identify ≈10% (7510/75,085) webpages as parking. These were discarded from further processing.

3.3.3 HCL: Hierarchical Clustering Module

To find clusters of related domain names associated with cross-channel abuse in a given epoch (time period, t), we follow a hierarchical clustering process. This process can be separated into three different levels. In the first level (NCL), we cluster together domain names based on the network infrastructure properties. In the second level (PCL), first level (NCL) clusters that satisfy a cardinality constraint (based on threshold λ) get further clustered according to the DNS volumetric popularity of the domains within it. In the third and final clustering step (ACL), second level (PCL) clusters that satisfy an entropy (flux) constraint (based on threshold ϵ) get further

separated based on the web content of each domain within it. This way, the entire process produces clusters of high quality at different levels which are then labeled by the attribution module (Section 3.3.4).

In order to execute these three different clustering steps, we employ the most common statistical features from the areas of DNS [54, 55, 56] and HTML [87] modeling. Similar hierarchical clustering techniques have been shown effective in multiple contexts [92, 72, 73]. Moreover, the clustering technique assumes that the input dataset is available as whole when the algorithm begins its task. It may be possible to create a more real-time and incremental version [91] of the algorithm using additional hardware resources [89, 90]. We leave this advancement for future work.

To be clear, these individual features may not be novel themselves. Rather, our goal is to show that already discussed features combined in this novel hierarchical clustering method can provide an efficient and effective attribution system for SMS-spam abuse. Next, we briefly discuss how we used these established statistical features in the context of the three modules of our system.

3.3.3.1 Network-based Clustering (NCL):

To compute network layer features in a given time epoch t, for each domain d in the domain set $\mathcal{D}$ under consideration, we compute two sets: (i) RHIP(d) which is a set of all IPs that have historically mapped to domain d, and (ii) RHDN(IP) which is the set of domains that have historically been linked with the IP in the RHIP set. This could also include domains that are not in $\mathcal{D}$. Using the collection of all domains $\mathcal{D}$, the $pDNS$ dataset and a specified epoch t, the network feature-based clustering submodule generates a matrix $A_{m \times n}$ where $m = |\mathcal{D}|$ represents the total number of domains and $n = |\bigcup_i RHIP(d_i)|$ represents the total number of IPs historically associated with all domains in $\mathcal{D}$ during an epoch t. The matrix A is computed as follows,

$$A_{i,j} = \begin{cases} \frac{H(d_i)}{|RHDN(ip_j)|} & \text{if } ip_j \in RHIP(d_i) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $i \in \{0, 1, \ldots |\mathcal{D}| - 1\}$ and $j \in \{0, 1, \ldots | \bigcup_i RHIP(d_i)| - 1\}$. Also, $H(d) = -\sum_{k \in C(d)} p_k * log_2(p_k)$, where $C(d)$ represents the unique set of characters in domain name d and p_k represents the probability of the occurrence of a given character in the domain name. Thus, $H(d)$ gives us the entropy of the name of domain d based on relative character frequencies. The inclusion of the entropy factor in the numerator increases the confidence of producing high quality clusters given the frequent use of DGAs [57, 101] by adversaries.

Finally, we use Singular Value Decomposition (SVD) [100] to reduce the dimensionality of the sparse matrix $A_{m \times n}$ to $A_{m \times \widetilde{n}}$ where $\widetilde{n} < n$. The network clustering module then uses the X-Means clustering algorithm [83] to cluster domains having similar network-level properties.

3.3.3.2 Popularity-based Clustering (PCL):

Sometimes, network level properties may be insufficient to distinguish between unrelated domains, leading to the formation of large clusters. We will see this in Section 3.4.2.1. Popularity based clustering uses features extracted from observing the popularity of domain names as measured by the number of the successful DNS resolutions to it within the epoch t. This in turn gives us a lower bound on the number of visits potentially made to the domain name via clicking on a URL embedded in an SMS message. It is computed using the information gathered in the passive DNS dataset. Let Lookup(d, dt) be a function that returns the number of lookups (or in other words, successful DNS resolutions) for domain d on a given date dt. And let C be the set of clusters produced by NCL. Using the pDNS data collection and a specified epoch t, the popularity cluster submodule builds matrices $B_{p \times q}(c_r)$ $\forall c_r \in C$ s.t.

$|c_r| \geq \lambda$, $r \in \{0, 1, \ldots |C| - 1\}$ where λ is a provided threshold and $|C|$ is the number of clusters produced by NCL. Here, $p = |c_r|$, the number of domains in a cluster from NCL and q are the total dates in a given epoch. The matrix B is computed as follows, $B_{i,j}(c_r) = Lookup(d_i, dt_j)$ where d_i is a domain name and dt_j is a date in epoch t and c_r is a NCL cluster. The intuition behind this matrix follows from the work by Antonakakis et al. [55] which aims to measure the volumetric DNS request patterns to domain names over time, within a NCL cluster (in our case).

Similar to the NCL module, each matrix is dimensionally reduced using SVD followed by X-Means clustering algorithm to cluster domains having similar popularity levels. Therefore, at the end of PCL, we have: (i) smaller clusters from NCL that had sufficient network level information ($|c_r| < \lambda$), and (ii) PCL (sub)-clusters from the larger NCL clusters that required the additional popularity information for further refinement.

3.3.3.3 Application-based Clustering (ACL):

To further refine and resolve any remaining confusion between domain names after PCL, we proceed to a final clustering step that aims to group together domain names with similar domain structure and web content. To cluster similar domains based on their structure, we compute the standard deviation σ of the entropy of domain names in a cluster produced after the PCL module. Let T represent the set of domains in a PCL cluster and $H(T)$ be the set of entropies associated with domain names in T. If $\sigma(H(T)) \geq \epsilon$, i.e., the standard deviation in the entropy of the domain names in the cluster is greater than the threshold ϵ, we apply application based clustering to a PCL cluster. Again, the motivation behind using entropy as a metric to assess the quality of clusters is similar to its purpose during NCL.

Once the clusters requiring application based clustering are identified, we use features extracted from the full HTML source of the web pages associated with domains.

Note that there could be multiple and different sources of web pages associated with a certain domain. We use the timestamp of the complaint associated with domains to identify relevant HTML sources in a given epoch. Once we have the domains and their corresponding HTML content, we compute TF-IDF statistical vector on the bag of words on each cluster c [87]. Since the matrix is expected to be quite sparse, the application cluster submodule performs dimensionality reduction using SVD. Once we have the reduced application based feature vectors representing corresponding domains, this module uses the X-Means clustering algorithm to cluster domains hosting similar content.

3.3.4 AM: Cluster Attribution Module

The cluster attribution module is used to label clusters with keywords that are representative of a campaign's theme. To do this, we leverage the observation that a majority of the domain names involved with cross-channel abuse, despite being auto-generated using domain generation algorithms (DGAs) [57, 101] , have certain keywords in the domain name itself that are relevant to the theme of a campaign. In other words, the domain names are not completely random. The aim is to lure the victim into visiting these domains via their smartphones and a well designed domain name increases the odds of clicking the URL. For example, domain names `yourfastcashsystem[dot]com`, `24hrpaysite[dot]com`, `target.com.ctarg[dot]com`, have keywords cash, pay and target respectively that give us useful clues to what the domain might pertain to.

Using this observation, we use the Viterbi algorithm [65] to filter the domain names in a given cluster to a sequence of words such as [`your`, `fast`, `cash`, `system`] in the case of `yourfastcashsystem[dot]com` and [`24`, `hr`, `pay`, `site`] in the case of `24hrpaysite[dot]com`. More formally, let C be a cluster produced after the entire clustering process and let D be the set of domains in the cluster. For each domain

$d \in D$, we create a set $U(d)$ that consists of all the parts of the domain name d except the effective top level domain (eTLD) (e.g. U('abc.example.com') = {abc, example}). Next, we compute the set of words $W(U(d))$ using the Viterbi algorithm. Therefore, W(U('abc.example.com')) = {example} since 'abc' is not a valid English word. Using W, we increment the frequency counter for the word 'example' in a cluster specific dictionary. In this manner, after iterating over all domains in the cluster, we get a keyword to frequency mapping from which we pick the top most frequent word(s) to attribute the cluster.

3.4 Results

In this section, we begin by describing the data collected and used in CHURN for SMS-spam attribution. We then dive deeper into both CHURN's clustering results and the attribution accuracy of the system.

3.4.1 Datasets

CHURN starts with an SMS-spam repository we developed from the sources mentioned in Section 3. It had ≈8.32 million SMS-spam reports. The data collection module used the domain names found in these reports to collect surrounding pDNS, HTML and domain blacklist information using passive and active crawling methods. All these datasets were continuously gathered over a period of four years and eight months, starting in January 2011 and ending in August 2015, ensuring an overlapping time period.

The pDNS crawler was able to observe and record DNS Resource Records (RRs), which gives us a temporal mark between a domain name and an IP address when the SMS-spam was active. We collected 17,528 unique fully qualified domain names, 23,037 distinct IP addresses and 56,940 unique RRs related to the cross-channel abuse. Regarding the HTML datasets around this SMS spam abuse, we were able to download 67,575 distinct pages with the corresponding HTML source code. We summarize

Table 3: Summary of collected datasets.

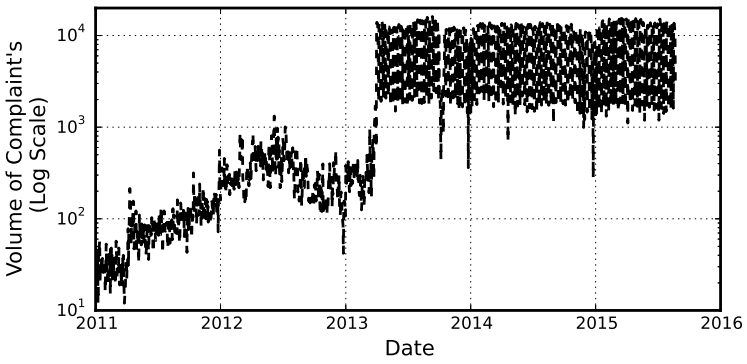
Epoch	RRs (Domain, IP) tuples	Domains (FQDN)	IPs (Hosts)	HTML sources	Complaints
Jan - Dec 2011	17,291	6,159	10,537	16,492	30,973
Jan - Dec 2012	17,316	7,846	8,218	16,321	125,960
Jan - Dec 2013	18,374	7,682	8,793	15,553	2,504,836
Jan - Dec 2014	22,426	7,438	8,858	15,334	3,286,988
Jan - Aug 2015	10,165	5,067	5,627	3,875	2,371,417
Total:	56,940	17,528	23,037	67,575	8,320,174

all this information across different epochs in Table 3.

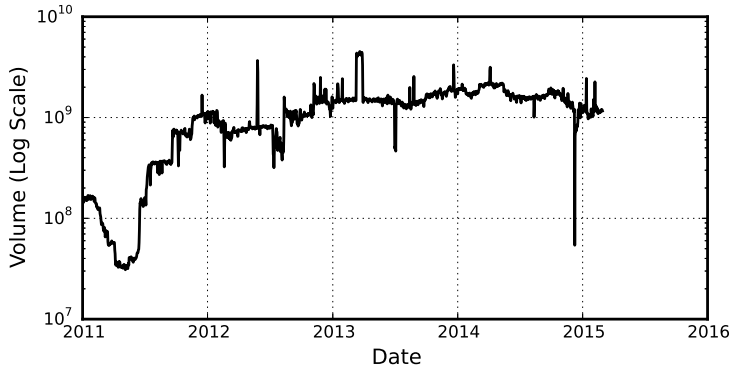
3.4.1.1 Temporal Characteristics of Cross-Channel SMS-Spam

Figure 4(a) shows the number of daily SMS complaint reports retrieved and analyzed by our system. Although there are fluctuations in the number of daily complaints, the overall volume of such complaints steadily increased over time. We suspect that the sudden surge in the number of complaints received in early 2013 is due to both a proactive effort by both FTC (and other regulatory parties) to encourage people to report such spam and also an increase in the awareness among consumers of the available reporting tools. The period between mid-2013 to mid-2015 shows a relatively steady volume of SMS-spam reports with only marginal increase in the number of daily complaints. This signals that the more dominant spam campaigns had stablized during this time period. In addition, it is also possible that the number of consumers willing to report such spam had reached a saturation point. Finally, Figure 4(b) shows the daily aggregated DNS lookup volume to SMS-spam domains based on data collected from a large passive DNS repository. We clearly see an uptake and a steady DNS lookup volume over time, showing that the cross-channel SMS based abuse is a persisting phenomenon.

Lifetime of SMS-Spam Domains Figure 5 shows the empirical cumulative distribution function (eCDF) of the lifetime of all domains seen in the campaigns.



(a) Number of daily complaints from both smswatchdog.com and FTC complaints.



(b) Daily aggregated passive DNS lookup volume trend for cross-channel spam domains.

Figure 4: Temporal characteristics of collected datasets.

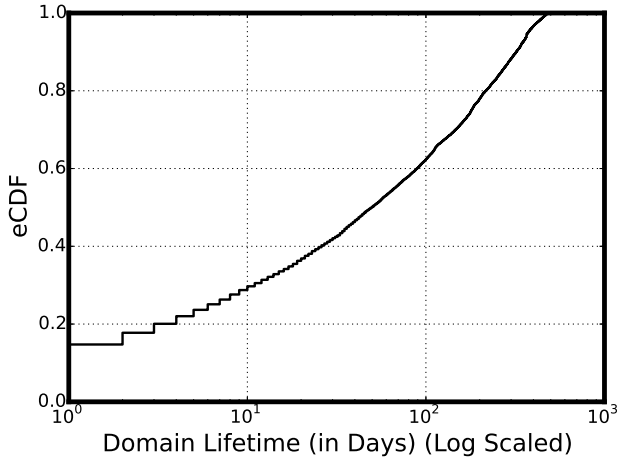


Figure 5: The eCDF of the lifetime of all domains showing long-lived SMS-spam domains.

The lifetime of a domain is derived by using the timestamp of the first and last seen DNS resolution to a particular domain. We observe that $\approx 30\%$ of the domains had a lifetime of less than 10 days, close to $\approx 30\%$ of domains had a lifetime between 10 and 100 days and the remaining $\approx 40\%$ had a lifetime between 100 and 480 days. This indicates that cross-channel spam domains are alive for much longer periods compared to traditional spam abuse, and even certain type of agile botnet abuse such as fast-flux networks [82]. To better study the evolution of SMS-spam abuse, in the remainder of the chapter, we break and analyze the datasets into yearly epochs.

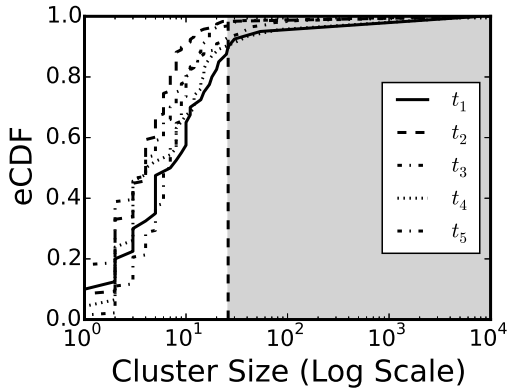
3.4.2 Clustering Results

Given a time period or an epoch and a set of domains, CHURN processes them in the hierarchical way as described in Section 3.3.3. We discuss the clustering results at various levels next.

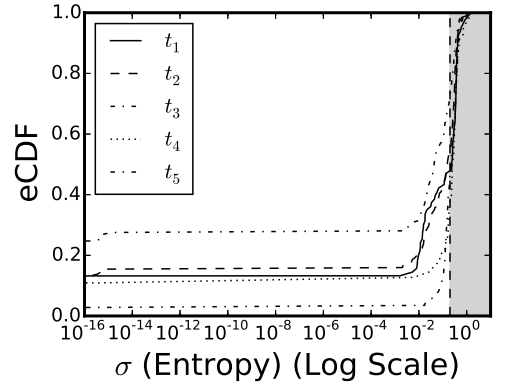
3.4.2.1 Clustering Network & Application Level Information

Figure 6(a) shows the empirical cumulative distribution of the cardinality (size) of the clusters produced after the network based clustering (NCL) step. Most of the clusters at this level contain few domains, but there exist some clusters that are quite large. We observed that up to 10% of the clusters produced during network level clustering had a cardinality ≥ 25, with one cluster being as large as almost half the number of domains under consideration. For these large clusters we leverage the domain popularity information to further break them down during the popularity based clustering (PCL) phase. By setting $\lambda = 25$, we were able to identify clusters to be processed by the popularity clustering submodule.

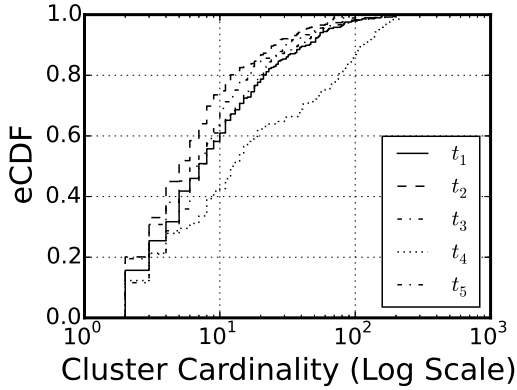
Once we have clusters from the NCL and PCL phases, the resulting clusters with disparate domain names are further refined using application level clustering (ACL). This is necessitated for some large clusters produced in the PCL module. Figure 6(b) shows the eCDF of the standard deviation (σ) in entropy of domain names for all



(a) eCDF of the cardinality of the clusters produced in the NCL module. Clusters with cardinality $\geqslant \lambda = 25$ (shown as vertical line $x = 25$) are processed further.



(b) eCDF of the standard deviation (σ) of entropy of domain names for clusters after the PCL module. Clusters with $\sigma \geqslant \epsilon = 0.2$ (shown as vertical line $x = 0.2$) are processed further.



(c) eCDF of the cardinality of all the clusters produced after all modules (NCL, PCL and ACL) for five different epochs.

Figure 6: HCL Thresholds

Table 4: Representative sample of attributed clusters at various levels of the clustering hierarchy. Apart from the above and the case studies, we discovered campaigns related to selling drugs, adult content, free cruises, fake deals and many more.

Cluster Level	Domain-(FQDN)	Label(s)	Epoch	Sample Domains
3	8	wire, deposit	2011	`wire600.com`, `deposit1500.com`
1	23	buy, best	2012	`bestbuy.com.bexy.biz`, `bestbuy.com.bwty.biz`
2	20	phone	2012	`mobiletestandkeep.com`, `iphone5tryout.com`
3	58	cash	2013	`startcreatingcash.com`, `trackingyoursuccess.com`
1	4	news	2014	`cnbcnews29.com`, `cnbcnews34.com`
3	129	loans, day, pay	2015	`instanteasyloans.co.uk`, `checkonlinepaydayloans.com`

clusters thus produced, differentiated based on epoch. Selecting as threshold $\epsilon = 0.2$, we were able to mark up to 60% of the clusters for further processing by the ACL module. Note that both the parameters λ used in PCL and ϵ used in ACL could be set according to the operator's needs. The application level clustering module gave us fine-grained clusters of very good quality with the largest cluster consisting of 201 domains across all epochs. Figure 6(c) shows the eCDF from the distribution of final cardinalities of all the clusters produced after all modules (NCL, PCL and ACL).

3.4.2.2 AM Results

The attribution module (AM) is used to label the clusters with keywords based on the domain name patterns. For illustration, Table 4 shows a sample output from this module. It can be seen that domains from certain campaigns can be attributed immediately after the NCL module. Some, however, are attributed after the PCL module and others after the ACL module. Figure 7 graphically depicts all the attributed clusters in our study at different levels for epoch t_2 (2012) as a radial dendogram plot. The center represents all the domains under consideration and the concentric circles represent the cluster labels at each level starting from NCL (level 1), to PCL (level 2) and ACL (level 3), as we move outward radially. This shows that some campaigns can be identified just by using network features, while others require a combination of network, popularity and application features.

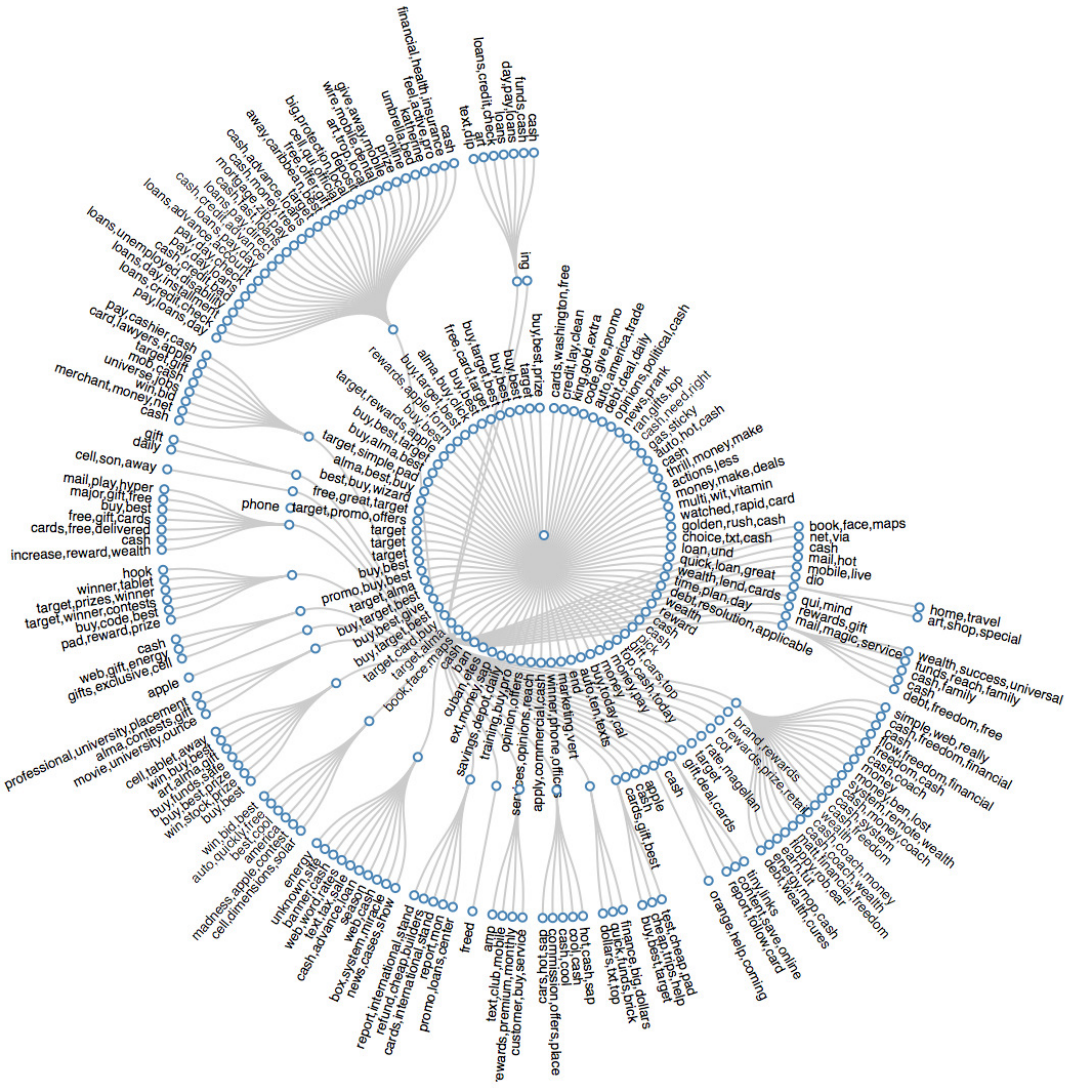


Figure 7: A radial dendrogram plot illustrating the output from the hierarchical clustering module for a single epoch.

3.4.2.3 Evaluation

To evaluate the output of CHURN and validate our results, we created ground truth data by labeling domains with group labels. Each group label represents a campaign. We made the judgement of assigning a specific group label to a domain based on looking at the domain names and loading up their associated webpages in a browser. Our experiment consisted of six group labels corresponding to the Bestbuy, Target, Walmart, Financial Freedom, Payday and News campaigns depicted as Group 1-6 in

Table 5: CHURN evaluation based on ground truth with different system parameter settings across all epochs.

		Group1	Group2	Group3	Group4	Group5	Group6	Total	Parameter Setting
1.	✓	77	65	14	277	205	12	650	$\lambda = 25$ & $\epsilon = 0.2$
	✗	0	0	0	0	2	1	3	
2.	✓	76	57	14	257	192	12	609	$\lambda = 2$ & $\epsilon = 10^{-12}$
	✗	1	8	0	20	15	1	44	
3.	✓	67	54	10	208	155	10	504	$\lambda = 2$ & $\epsilon = 2$
	✗	10	11	4	69	52	3	149	
4.	✓	64	35	8	188	125	7	427	$\lambda = 10000$ & $\epsilon = N/A$
	✗	13	30	6	89	82	6	226	
Total		77	65	14	277	207	13	653	

that order. We were able to label 653 (3.7%) domains in total to help us validate our results.

Table 5 shows how the results from CHURN measured up against the labeled data. System parameters λ and ϵ are varied to show the different cases. When λ is set to a relatively large value (i.e., 10,000), the output from the HCL module of CHURN is reduced to just the output of the NCL module since condition for PCL processing is never satisfied. The fourth threshold configuration shows that 427 out of the 653 domains were correctly attributed by CHURN using this setting. In the case when λ is set to a relatively small value (i.e., 2) and ϵ is set to a relatively large value (i.e., 2), the output from the HCL module of CHURN is reduced to output produced from applying the NCL and PCL modules sequentially but skipping the ACL module altogether. The third configuration shows that we attributed 504 out of 653 domains correctly using this setting.

Next is the case where λ and ϵ both are relatively small (i.e., 2 and 10^{-12} respectively). Such a setting results in all the modules NCL, PCL and ACL being serially applied to all clusters and domains without exception. This second configuration run shows that the number of correctly attributed domains increases from 609 to 653 domains. Finally, when λ and ϵ are set to 25 and 0.2 respectively, based on the justification presented in Section 3.4.2.1, NCL, PCL and ACL are applied to domains and clusters depending on the condition(s) being satisfied. This resulted in

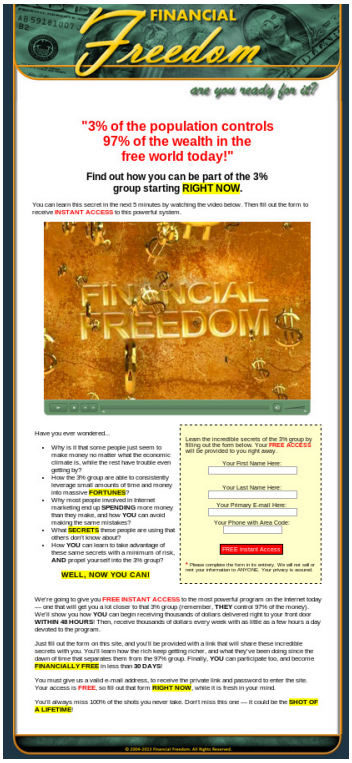
a marked improvement with 650 out of 653 domains being correctly attributed. The first configuration shows the results using this setting.

3.5 Case Studies

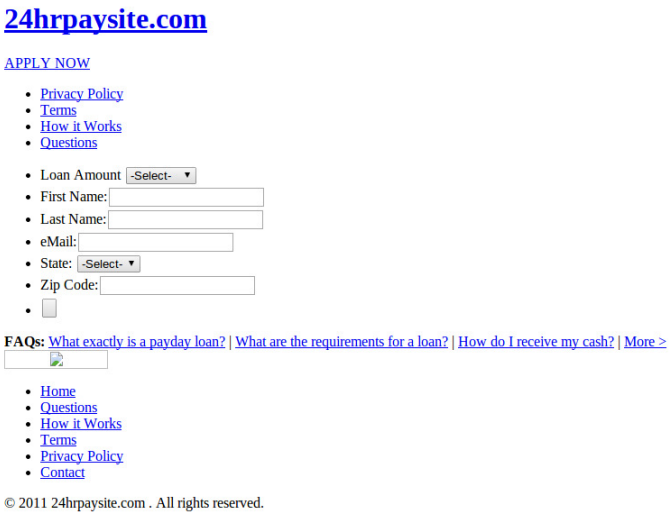
After CHURN's attribution module generates labels for clusters, these clusters and their associated labels are used to identify and group domains that are part of the same scam campaign. We present case studies for three of the most prominent campaigns (Figure 8) that are known SMS scams. As a general takeaway across all three case studies, we observed that the domains supporting the scams were hosted in diverse but few IP locations and for a long period of time. While the distributed infrastructure ensures reliability, the long term activity behind the domain names suggests the relative ineffectiveness of defenses against these social engineering cross-channel attacks compared to similar attacks via the internet channel.

Financial Freedom: Upon landing on the Financial Freedom web page an embedded video explains the purported benefits of enrolling into the program. The victim is asked to provide her personal information for 'Free Instant Access' to the program. The scam targeted consumers who are financially weak and looking for a solution to credit card debt problems. In our dataset, this scam consisted of 277 FQDNs (e.g. `morefreedomforall[dot]com`) and 187 IPs belonging to 49 distinct /24 subnetworks. None of the domains in this scam were seen in domain blacklists and the domains ended up being clustered in the ACL module. Figure 11(a) shows that the campaign used dedicated infrastructure to operate in a stealthy mode thus surviving for a long time, as can be seen in Figure 9(a), 10(a). Legal proceedings of a law suit initiated against the perpetrators of this scam can be found here [15].

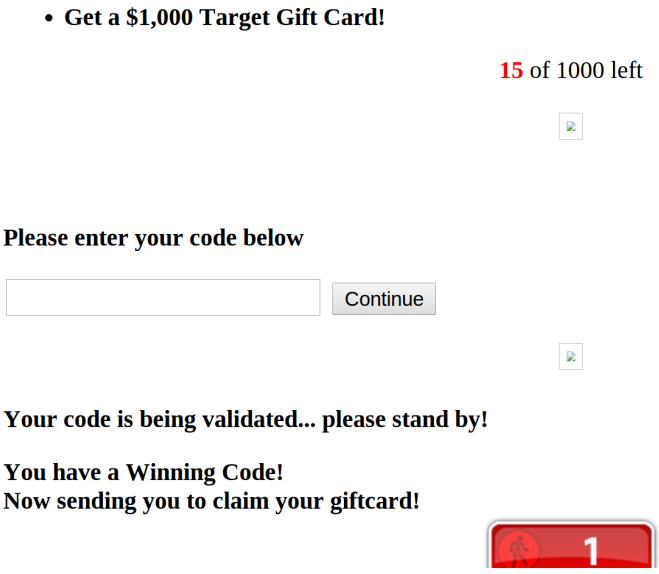
Payday: Payday loan is a short term, high interest cash advance that has been banned in many states in the United States, and the Federal Trade Commission (FTC)



(a)



(b)



(c)

Figure 8: Three Campaigns: Financial Freedom, Payday and Gift Card. For each we show (8(a)–8(c)) web pages rendered on a mobile browser.

has issued warnings regarding it [16]. For example, in one instance the defendants' online contract stated that a $300 loan would cost $390 to repay, but the defendants then charged consumers $975 to repay the loan. This is a case of obscuring the 'Terms of service' specified on the site, which make it hard for the victim to realize they are being scammed. The scam works by sending a victim a SMS message with a URL. Upon clicking the URL, the victim is asked to enter personal information, phone number, and loan amount to proceed further.

A particular online payday loan campaign was clustered in our SMS spam dataset consisting of 207 unique domains; hosted in 212 unique IP addresses; belonging to 142 distinct /24 subnetworks. 68 out of 207 such domains were part of the `.co.uk` TLD. Eight domains in this scam were seen in PBL and they were mainly clustered by the ACL module. Figure 9(b), 10(b) shows that despite the warnings by consumer protection authorities (especially in the USA), this scam has survived and continues to victimize consumers. In addition to this, Figure 11(b) shows the stability behind the network infrastructure used to support the scam domains.

Giftcard: In this case study, the scam works by sending the victim a SMS message with a URL and a code. Upon clicking the URL, the victim is asked to enter his/her personal details including phone number followed by entering the code in order to receive a fake free gift card from the associated brand (e.g., Target, Bestbuy, Walmart etc.). Thereafter, victims were told to sign up for more than a dozen risky trial offers, none of which were free, to qualify for the promised 'free' gift card. In many cases, the correct code confirmed to the gift card scam operators that the mobile number is indeed active and they use this entry as a pretense to falsely subscribe the victim's mobile number to premium rate services.

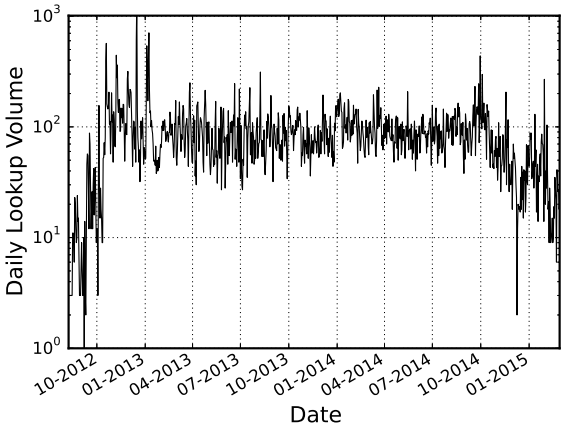
The campaign consisted of 207 domains and 215 IPs belonging to 85 distinct /24 subnetworks. Four domains under this scam were seen in PBL and the domains were

mostly clustered in the NCL module. This campaign was mostly active during two distinct time periods in 2012 and 2013, as can be seen in Figure 9(c). The resurgence of the campaign the second time coincides with the shopping/holiday season between November 2012 and January 2013 where a lucrative deal for a gift card is more likely to catch the victim's attention. Figure 10(c) shows that ≈45% of the domains had a lifetime of less than 10 days, ≈45% were active between 10-100 days and the remaining ≈10% of the domains were relatively long lived. We found that out of 207 domains, many of them were well crafted 4LDs (4th level domains), named after specific brands such as BestBuy (114), Target (77) or Walmart (16) e.g. `target.com.tthg[dot]biz`. We also noticed that the domains hosting these web pages have very similar layout, structure and content. The majority of the scam domains had a relatively shorter lifetime and were more agile in using their network resources.

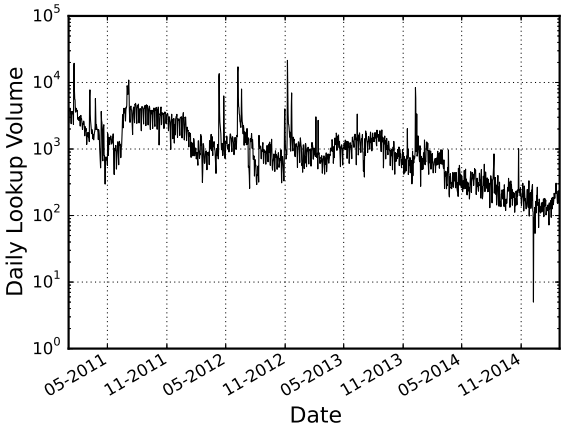
The FTC pressed charges against the perpetrators of the campaign for illegally sending ≈42.5 million text messages to consumers containing bogus offers for 'free' Gift Cards. These charges were publicly reported to be settled in September 2013 [45]. This is reflected in Figure 9(c), where we see very few to no lookups during the second half of 2013.

3.6 Summary

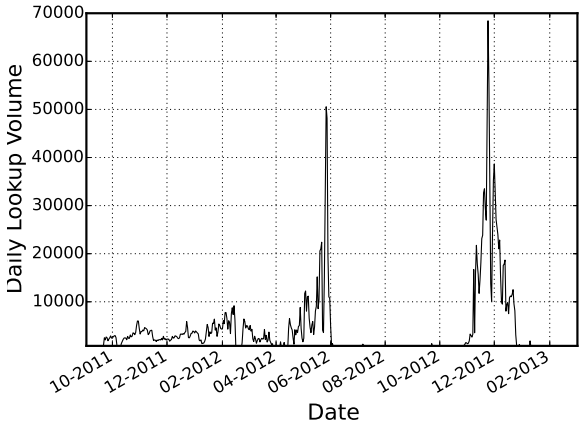
In cross-channel abuse, SMS-spammers are able to exploit the ubiquity of mobile devices and trust in the telephony channel to craft attacks that could be more successful than spam on the Internet channel alone. Such illicit activities have become a serious problem, with several reported scams that have lasted for several years. Using data from multiple sources, we seek to attribute cross-channel abuse to the Internet infrastructure that facilitates it.



(a)

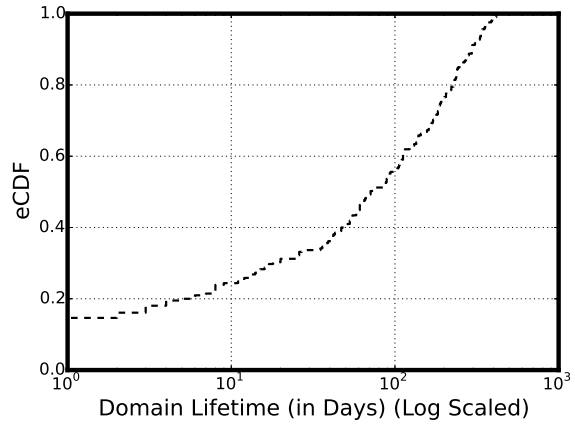


(b)

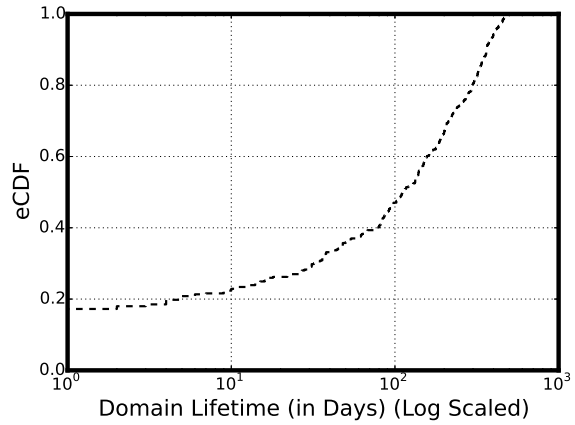


(c)

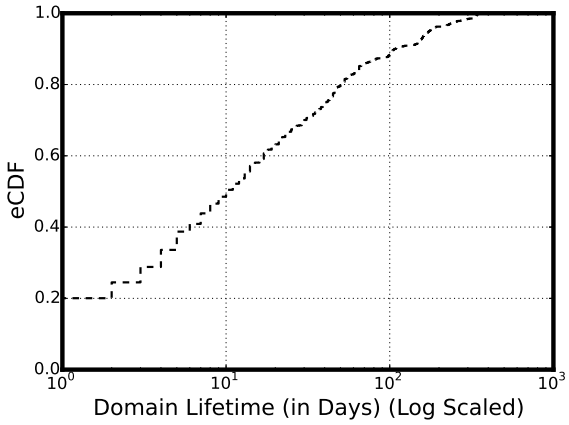
Figure 9: Three Campaigns: Financial Freedom, Payday and Gift Card. For each we show (9(a)–9(c)) daily lookup volumes according to our pDNS database.



(a)

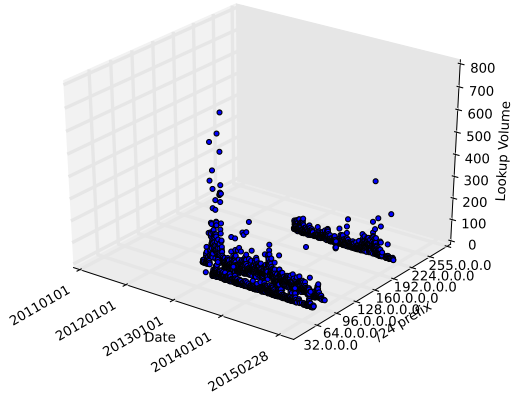


(b)

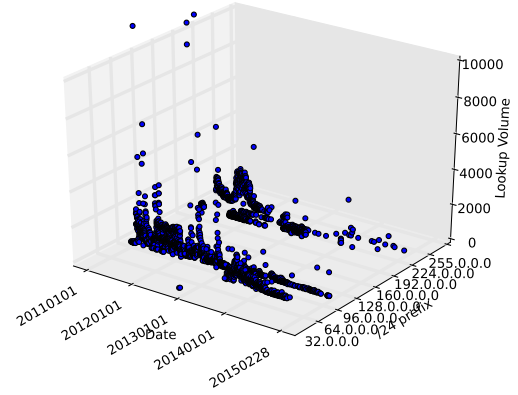


(c)

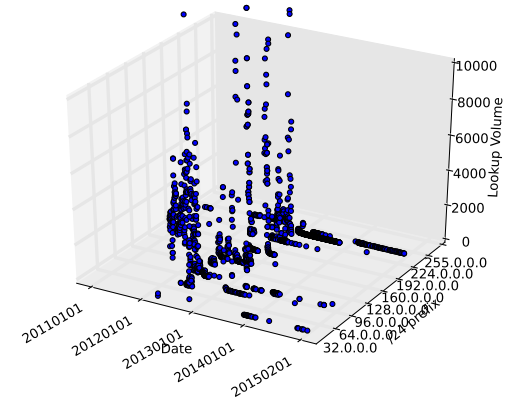
Figure 10: Three Campaigns: Financial Freedom, Payday and Gift Card. For each we show (10(a)–10(c)) eCDF of the lifetime of the domains seen.



(a)



(b)



(c)

Figure 11: Three Campaigns: Financial Freedom, Payday and Gift Card. For each we show (11(a)–11(c)) 3D view of campaigns based on time, popularity and network infrastructure (IPs binned by /24 prefix).

CHAPTER IV

X-TSS: MEASURING SEARCH AND AD-BASED CROSS-CHANNEL ABUSE IN TECHNICAL SUPPORT SCAMS

Technical Support Scams (TSS), which combine online abuse with social engineering over the phone channel, have persisted despite several law enforcement actions. The tactics used by these scammers have evolved over time and they have targeted an ever increasing number of technology brands. Although recent research has provided important insights into TSS, these scams have now evolved to exploit ubiquitously used online services such as search and sponsored advertisements served in response to search queries. We use a data-driven approach to understand search-and-ad abuse by TSS to gain visibility into the online infrastructure that facilitates it. By carefully formulating tech support queries with multiple search engines, we collect data about both the support infrastructure and the websites to which TSS victims are directed when they search online for tech support resources. We augment this with a DNS-based amplification technique to further enhance visibility into this abuse infrastructure.

By analyzing the collected data, we provide new insights into search-and-ad abuse by TSS and reinforce some of the findings of earlier research. Further, we demonstrate that tech support scammers are (1) successful in getting major as well as custom search engines to return links to websites controlled by them, and (2) they are able to get ad networks to serve malicious advertisements that lead to scam pages. Our study period of approximately eight months uncovered over 9,000 TSS domains, of both passive and aggressive types, with minimal overlap between sets that are reached via

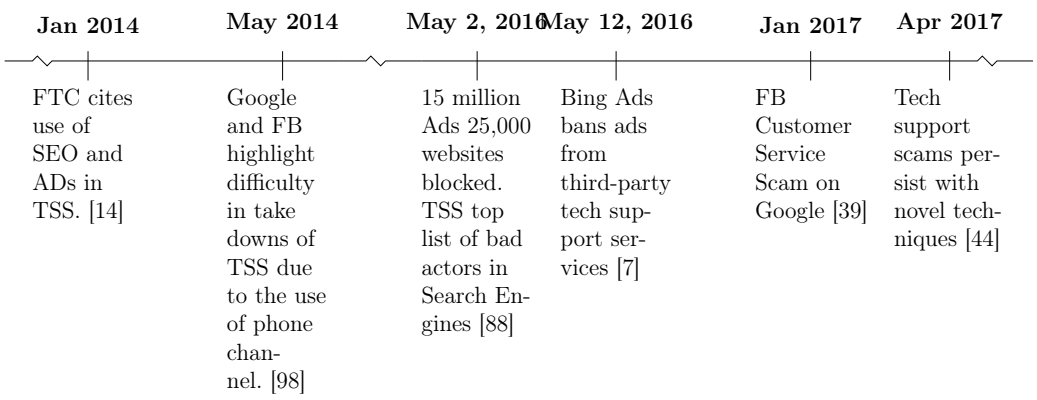


Figure 12: Timeline of some news events related to search-based technical support scams (TSS).

organic search results and sponsored ads. Also, we found over 2,400 support domains which aid the TSS domains in manipulating organic search results. Moreover, to our surprise, we found very little overlap with domains that are reached via abuse of domain parking and URL-shortening services which was investigated previously. Thus, investigation of search-and-ad abuse provides new insights into TSS tactics and helps detect previously unknown abuse infrastructure that facilitates these scams.

4.1 Context and Contributions

The *Technical Support Scam* (TSS), in which scammers dupe their victims into sending hundreds of dollars for fake technical support services, is now almost a decade old. It started with scammers making cold calls to victims claiming to be a legitimate technology vendor but has now evolved into the use of sophisticated online abuse tactics to get customers to call phone numbers that are under the control of the scammers.

In their pioneering research on TSS [80], Miramirkhani et. al. explored both the web infrastructure used by tech support scammers and the tactics used by them when a victim called a phone number advertised on a TSS website. They focused on TSS websites reached via malicious advertisements that are served by abusing domain parking and ad-based URL shortening services. Although their work provided

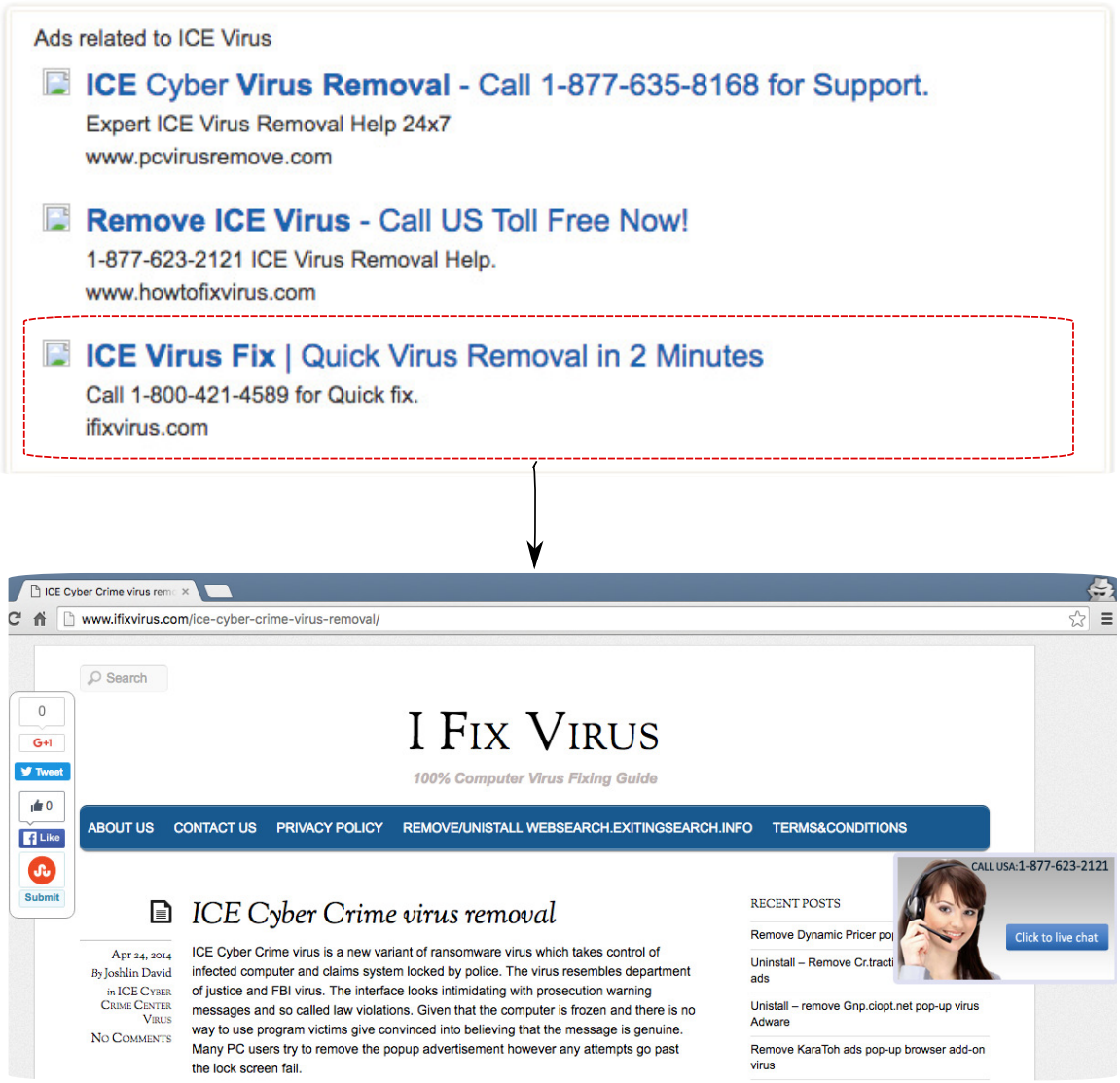


Figure 13: Goentry.com search results on July 1st, 2016.

important insights into how these services are abused by TSS, it has recently become clear that tech support scammers are diversifying their methods of reaching victims and the ways with which they convince these victims to call them on their advertised phone numbers.

Specifically, recent reports by the FTC and by search engines vendors suggest that scammers are turning to search engine results and the ads shown on search-results pages as novel ways of reaching victim users [14, 39, 98]. These new channels not only allow them to reach a wider audience but also allow them to diversify the ways with

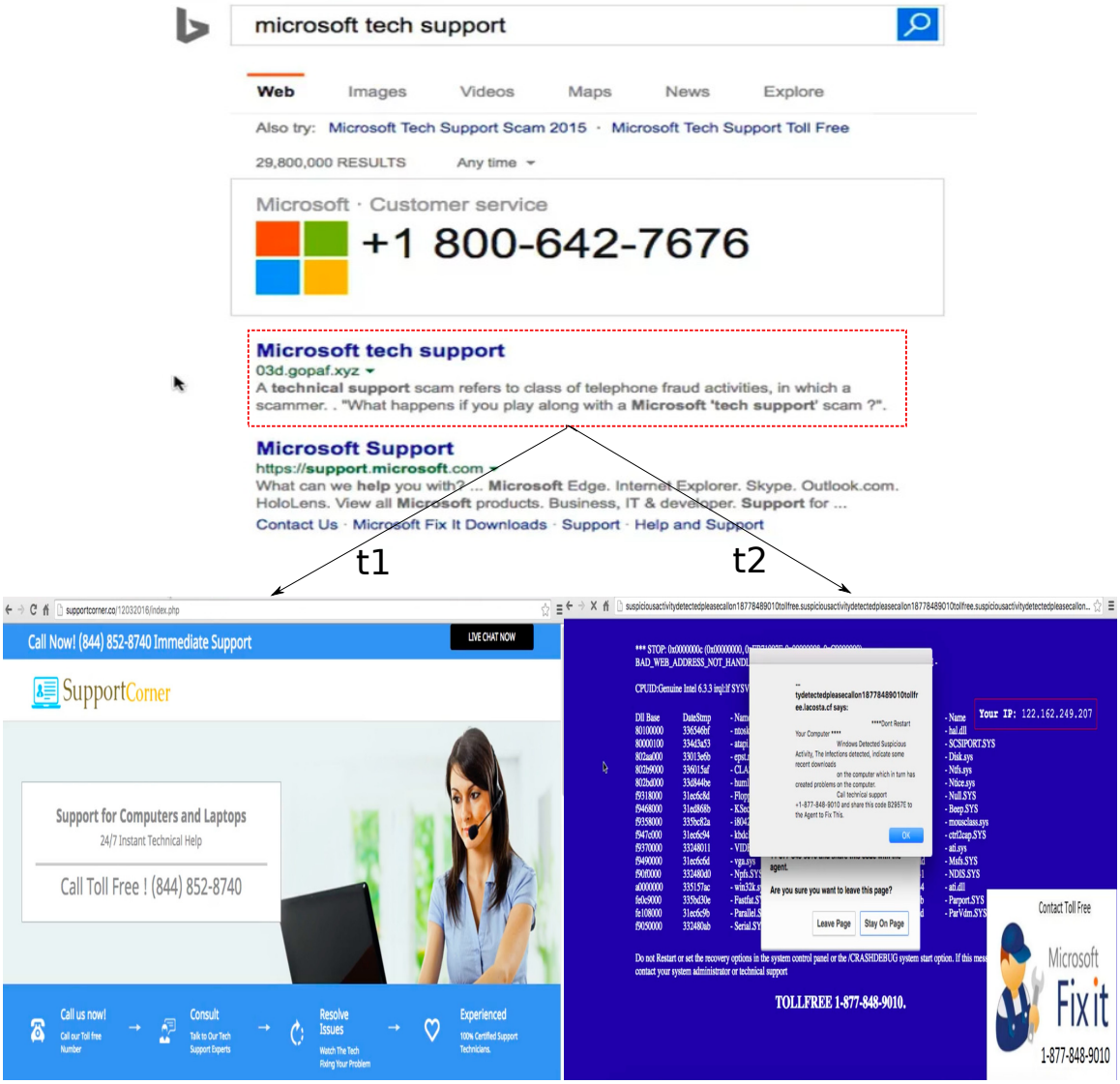


Figure 14: Bing.com search results on Feb 2nd, 2017.

which they attempt to convince users to call them. As shown in Figure 12, several actions have been taken to stop TSS but these scams continue to adapt and evade both law enforcement and technical safeguards.

In this chapter, we perform the first systematic study of these novel search-and-ad abuse channels. We develop a model for generating tech-support related queries and use the resulting 2,600 queries as daily searches in popular and less popular search engines. By crawling the organic search results and ads shown as a response to our queries (note that we follow a methodology that allows us to visit the websites of ads

without participating in click-fraud), we discover thousands of domains and phone numbers associated with technical support scams. In addition to the traditional *aggressive* variety of technical support scams (where the pages attempt to scare users into calling them), we observe a large number of *passive* technical support scam pages which appear to be professional, yet nevertheless are operated by technical support scammers (Figures 13 and 14 show examples of such scams). Using network-amplification techniques, we show how we can discover many more scam pages present on the same network infrastructure, and witness the co-location of aggressive with passive scam pages. This indicates that a fraction of these aggressive/passive scams are, in fact, controlled and operated by the same scammers. We also discover that the lifetime of passive scam pages is significantly larger than aggressive scam pages and find that our collected scams have little-to-no overlap with the scams identified by Miramirkhani et al.'s system during the same period of time. This indicates that our system reveals a large part of the TSS ecosystem that remained, up until now, unexplored.

Our main contributions are the following:

- We design the first search-engine-based system for discovering technical support scams, and utilize it for eight months to uncover more than 9,000 TSS-related domains and 3,365 phone numbers operated by technical support scammers, present in both organic search results as well as ads located on search-results pages. We analyze the resulting data and provide details of the abused infrastructure, the SEO techniques that allow scammers to rank well on search engines, and the long-lived *support* domains which allow TSS domains to remain hidden from search engines.

- We find that scammers are complementing their aggressive TSS pages with passive ones, which both cater to different audiences and, due to their non-apparent malice, have a significantly longer lifetime. We show that well-known

network amplification techniques allow detection systems to not only discover more TSS domains but to also trace both aggressive and passive TSS back to the same actors.

- We compare our results with the ones from the recent TSS study of Miramirkhani et al. [80] and show that the vast majority of our discovered abusive infrastructure is not detected by prior work, allowing defenders to effectively *double* their coverage of TSS abuse infrastructure by incorporating our techniques into their existing TSS-discovering systems.

4.2 Methodology

We utilize a data-driven methodology to explore TSS tactics and infrastructure that is used to support search-and-ad abuse. To do this, we search and crawl the web to collect a variety of data about TSS websites, and use network-level information to further amplify such data. Our system, which is shown in Figure 15, implements TSS data collection and analysis functions, and consists of the following six modules:

1. The *Seed Generator* module generates phrases that are likely to be used in search queries to find tech support resources. It uses a known corpus of TSS webpages obtained from Malwarebytes [32] and a probabilistic language modeling technique to generate phrases that serve as input to search queries.

2. Using search phrases, the *Search Engine Crawler (SEC)* module mines search engines including popular ones such as Google, Bing, and Yahoo! for technical support related content appearing via search results (SRs) and sponsored advertisements (ADs). We also mine a few obscure ones such as goentry.com and search.1and1.com that we discovered are used by tech support scammers. The SR and AD URIs are candidates for active crawling.

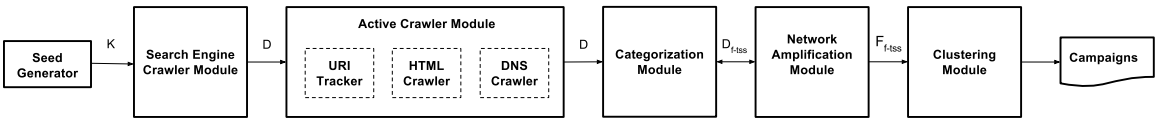


Figure 15: X-TSS: The Cross-Channel TSS threat collection and analysis system.

3. The *Active Crawler Module (ACM)* then tracks and records the URI redirection events, HTML content, and DNS information associated with the URIs/-domains appearing in the ADs and SRs crawled by the SEC module.

4. *Categorization module* which includes a well-trained *Technical Support Content Classifier (TSSC)*, is used to identify TSS SRs and ADs using the retrieved content.

5. The *Network Amplification Module (NAM)* uses DNS data to amplify signals obtained from the labeled TSS domains, such as the host IP, to expand the set of domains serving TSS, using an amplification algorithm.

6. Lastly, using the information gathered about TSS domains, the *Clustering Module* groups together domains sharing similar attributes at the network and application level.

4.2.1 Search Phrase Seed Generator

Selecting appropriate queries to feed the search engine crawler module is critical for obtaining suitable quality, coverage and representativeness for TSS web content. To do this, we must generate phrases that are highly likely to be associated with the content shown or advertised in TSS webpages. Deriving relevant search queries from a context specific corpus has been used effectively in the past for measuring search-redirection attacks [74]. We use an approach based on joint probability of words in phrases in a given text corpus [78].

We start with a corpus of 500 known technical support scam websites from the Malwarebytes technical support (TSS) domain blacklist (DBL) [32], whose content

Table 6: Summary and examples of generated n-grams related to technical support scams.

n	# ngrams	Example English Phrase
1	74	virus
2	403	router support
3	1,082	microsoft tech support
4	720	microsoft online support chat
5	243	technical support for windows vista
6	72	hp printers technical support phone number
7	6	contact norton antivirus customer service phone number
Total	2,600 english phrases	

was available. We were able to find 869 unigrams or single words after sanitizing the content in the corpus for stop words. We further reduce the number of single words or unigrams by only considering words that appear in more than 10 websites. This leaves us with 74 unique words. Using the raw counts of unigrams, we compute the raw bi-gram probabilities of eligible phrases with the chain rule of probability. We then use the Markov assumption to approximate n-gram probabilities [33]. Once we have probabilities of all phrases up to n-grams, we use a probability threshold λ_n to pick phrases having probability of occurrence greater than the threshold for each value of n. In effect, we develop a language model pertinent to technical support scam websites.

Table 6 shows the total number of phrases found for different values of n and some examples of the phrases found. We restricted the value of n to 7, as the value of $n = 8$ did not yield any phrases that would be logical as search engine inputs to find online technical support scams. As we can see, $n = 3$ yields a lot of popular phrases used in online technical support scams. In total, we were able to identify 2600 English phrases that serve as search queries to the SEC module.

4.2.2 Search Engine Crawler (SEC) Module

The SEC module uses a variety of search engines and the search phrases generated from the TSS corpus to capture two types of listing: traditional search results, sometimes also referred to as organic search results, and search advertisements, sometimes also referred to as paid/sponsored advertisements.

Both Google [20] and Bing [9] provide APIs that can be used to get SRs. However, some of the search engines we considered did not have well documented APIs and vanilla crawlers are either blocked or not shown content such as ADs. In such cases, we automate the process using PhantomJS [35], a headless WebKit "scriptable" with a JavaScript API. It allows us to capture a search page with both SR and AD listings as it would be shown to a real user visiting the search engine from a real browser.

Once we have the raw page p from the search engine in response to a query q, we use straighforward CSS selectors to separate the SRs from ADs. A SR object typically consists of basic components such as the the SR title, the SR URI, and a short snippet of the SR content. An AD object too, typically consists of these components, i.e. the AD title, the advertiser's URI/domain name, and a short descriptive text. The advertiser also provides the URI the user should be directed to when the AD is clicked. In addition, an AD may also consist of an AD extension component which allows actions to be performed after the AD is rendered (e.g. call extensions that allow the advertiser to embed a phone number as a clickable call button). The main difference between the contents displayed in SRs and ADs is that the content shown in the former is what is seen by the search engine crawler whereas the content in the latter is provided directly by the advertiser. The SR/AD along with its components are logged into a database as a JSON object. The URI component of the ADs and SRs are then inserted into the ADC (AD crawling) and SRC (SR crawling) queues respectively, which then coordinate with the ACM to gather more information about them, as discussed next.

4.2.3 Active Crawler Module (ACM)

The ACM uses the ADC and SRC URI queues to gather more information relevant to an AD/SR. ACM has three submodules that keep track of the following information for each URI seen in the AD/SR: (i) URI tracking, (ii) HTML and Screenshot Capture, and (iii) DNS information. We now discuss each of the submodules corresponding to these.

URI Tracker: The purpose of the URI tracker is to follow and log the redirection events starting from the URI component seen in the AD/SR discussed in the previous module. Barring user clicks, our goal is to capture the sequence of events that a real-world user on a real browser would experience when directed to technical support scams from SR/AD results, and *automate* this process. Our system uses a combination of python modules PhantomJS [35], Selenium [36] and BeautifulSoup [6] to script a light-weight headless browser. Finally, to ensure wide coverage, we configure our crawlers with different combinations of Referer headers and User-Agents (we discuss the exact settings in Section 4.3). Next, we discuss briefly how automating URI tracking (and other related events) can pose ethical challenges in the case of ADs and how we handle them.

Mimicking AD Clicks: When a user clicks on an AD, the click triggers a sequence of events in which the publisher, AD network and advertiser are involved, before the user lands on the intended webpage associated with the AD. This can be attributed to the way monetization model behind ADs work [63]. For example, the domain name shown in an AD could be *gosearch770.xyz* while the source URI associated with it is `hXXp://54080586.r.msn.com/?ld=d3S-92s04zd0&u=www.gosearch770.xyz%2findex.php`. Clicking on the AD may result in the flow of money from the advertiser to the AD network and publisher depending on the charging model such as Cost-per-click (CPC) or Pay-per-click (PPC). Clearly, the intent of our automated crawlers is not to interfere with this monetization model by introducing extraneous

clicks. One alternative to actually clicking on the ADs and a way to bypass the AD network is to visit the advertiser's domain name directly, while maintaining the *Referer* to be the search engine displaying the AD. In theory, any further redirections from the advertiser's domain should still be captured.

We chose the strategy that follows the advertiser's domain while ensuring that the same path (URIs and domain names) that leads to the technical support scam webpage is followed as if we had clicked on the AD. To validate if this was a viable option while maintaining accuracy of the data collection process, we conducted a controlled experiment in which we compared a small number of recorded URI resolution paths generated by real clicks to paths recorded while visiting the advertiser's domain name directly. We did this for the same set of technical support ADs while keeping the same browser and IP settings. For a set of 50 fake technical support ADs from different search engines identified manually and at random, these paths were found to be identical. This gives us confidence that accurate URI tracking information can be collected for fake technical support ADs without affecting the originating AD networks. For SRs, we just simulate a click on the SR and follow the SR URI component of the SR object. Thus, the outcome of this submodule is the URI redirection path which includes the fully qualified domains (FQDNs) encountered and the method of redirection for both ADs and SRs.

HTML Crawler: The HTML crawler works in conjunction with the URI Tracker. This crawler captures both the raw HTML as well as visual screenshots of webpages shown after following the ADs and SRs. For each domain d and webpage p, in the path from an AD/SR to the final landing webpage, the crawler stores the full source html and an image of the webpage as it would have appeared in a browser, into a database. It uses a combination of the domain name and timestamp as identifiers for this data, so that it can be easily referenced when needed. The content generated from this module is used in various other modules/submodules in order to decide

the threat level of the AD/SR and whether it is a fake technical support AD/SR (Section 4.2.4); extract the toll-free number used (if any); and to cluster campaigns of technical support scams (Section 4.2.6).

Active DNS Crawler: For each domain, d, in the path from an AD/SR to the final landing domain, the active DNS crawler logs the IP address, ip, associated with the domain to form a (d, ip, t) triplet, based on the DNS resolution process at the time of crawling, t. This information is valuable for unearthing new technical support scam domains (Section 4.2.5) and in studying the network infrastructure associated with cross-channel technical support scams (Section 4.4).

4.2.4 Categorization Module

Although we input technical support phrases to search engines with the aim of finding fake technical support websites, it is possible and even likely that some SRs and ADs lead to websites that are legitimate technical support or even completely unrelated to technical support. To categorize all search engine listings obtained during the period of data collection, we first divide the URIs collected from both ADs and SRs into two high-level categories: TSS and Non-TSS, (i.e. those URIs that lead to technical support scam pages and those that lead to benign or unrelated pages). Within each category, we have subcategories: TSS URIs are further separated into those leading to aggressive TSS websites and those leading to passive TSS websites.

TSS Website Classifier: We determine an AD/SR as technical support scam or not based on the webpage content shown in the final landing domain corresponding to an AD/SR. We leverage the observation that a lot of fake technical support websites host highly similar content, language and words to present themselves [80]. This can be represented as a feature vector where features are the words and values are the frequency counts of those words. Thus, for a collection of labeled TSS and Non-TSS websites, we extract the bag of words after sanitization (such as removing stop words),

and create a matrix of feature vectors where the rows are the final landing domains and the columns are the text features. We can then train a classifier on these features which can be used to automatically label future websites.

To that effect, we built a model using the Naive Bayes classification algorithm with 10-fold cross validation on a set comprising of 500 technical support scam and 500 non-technical support scam websites identified from the first few weeks of ADs/SRs data. The training set is randomly selected and manually labeled. The selection consists of representative samples of different kinds of TSS webpages, both passive and aggressive types, along with Non-TSS webpages that were found among the search listings including benign or unrelated webpages. The performance of the classifier is captured in the ROC Curve shown in Figure 16. We see that a threshold of 0.6 yields to an acceptable true positive rate (TPR) of 98.9% and a false positive rate (FPR) of 1.5%. Moreover the area under the curve (AUC), which is a measure of the overall accuracy of the trained model, is 99.33% which gives us confidence that the technical vs. non-technical support webpage classification works well. The outcome at this stage, after running it over new and incoming AD/SR data, is a set of final landing TSS webpages originating from an AD/SR. To make sure, we are not including, genuine, popular and high reputation technical support service websites in our TSS dataset, eg. Best Buy's Geek Squad [18], we drop domain names (if any), appearing in the Alexa top 10,000 websites list [5].

Next, to separate TSS URIs into those leading to passive/aggressive websites, we use the presence of features extracted from the HTML of the landing TSS website. Aggressive TSS websites exhibit behavior that contributes to a false sense of urgency and panic through a combination of audio messages describing the problem and continuous pop-up messages/dialogue loops which can be detected using tags such as <audio>, window.alert(), window.confirm(), window.prompt() etc. On the other hand, passive TSS websites adopt the approach of seeming genuine. This is

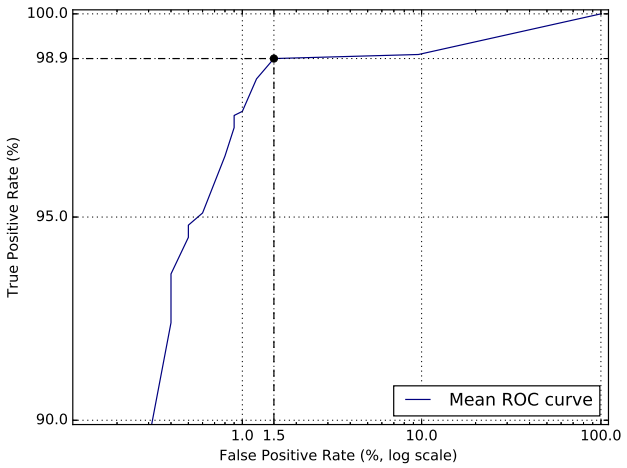


Figure 16: ROC Curve of the TSS Website Classifier on the training set.

accomplished by using simple textual content, certifications, seals, and other brand-based images. They often present themselves as official tech support representatives of large companies and, because of their non-apparent malice, pose new challenges for the detection of TSS [98].

To evaluate the performance of this TSS classifier, we sample data from the test AD/SR dataset. To verify actual TSS websites, we use Malwarebytes [32] TSS blacklist data as an independent source of ground truth. The blacklist consists of domain names and phone numbers that serve both passive and aggressive TSS. However, certain websites from the test set that are marked as TSS may not be listed in Malwarebytes. For these, we use a combination of manual analysis of the website content and IP co-location indicators to verify that the website is indeed associated with TSS. While aggressive TSS websites are easy to verify using characteristics of the website content itself, passive TSS websites require additional work for verification. Instead of calling the phone numbers listed on websites classified as passive TSS, we use IP co-location properties of passive TSS websites with known TSS websites as an indicator of scam. In Section 4.3.3, we show that indeed, some of the passive scams are operated out of the same IP infrastructure that runs the aggressive ones, giving us

Table 7: Confusion matrix for the TSS classifier on the testing set.

	Predicted TSS	Predicted non-TSS	Total
Actual TSS	196	4	200
Actual non-TSS	1	199	200
Total	197	203	

confidence in creating ground truth on passive TSS websites based on this feature. Using this strategy, we were able to evaluate the performance of the classifier on a groud truth dataset consisting of 200 TSS websites and 200 Non-TSS websites, sampled randomly from the test set. Among the TSS websites, there were 100 aggressive TSS websites and 100 passive TSS websites in the ground truth set. 114/200 TSS websites were verified via Malwarebytes and the remaining 86 websites were verified via IP-colocation property with a known TSS website. Table 7 shows the confusion matrix related to this experiment. The TSS classifier was able to achieve a reasonable 98% TPR and low 0.5% FPR on the testing set, thus validating the TSS website classification methodology. Also, there was 100% accuracy in distinguishing passive from aggressive TSS websites using the aforementioned heuristics. While we understand the limitations of this method of evaluating the testing set classification, we seek to improve and scale this experiment in the future using additional independent sources of ground truth data.

4.2.5 Network Amplification Module

Using search listings to identify active TSS websites works well for creating an initial level of intelligence around these scams. However, it may be possible to expand this intelligence to uncover more domains supporting TSS that may have been missed by our crawler (possibly because the domains were not actively participating in AD or SEO activity at the time). These domains may be dormant, perhaps, waiting to be circulated at a later stage. However, the give-away for these additional TSS

domains could be the sharing of network-level infrastructure with already identified TSS domains. Once we have a set of labeled final-landing domains, $\mathcal{D}_{f-tss}$, related to fake technical support websites originating from ADs/SRs, we leverage the properties of the Domain Name System (DNS) to find more fake technical support websites via an amplification process which works as follows.

A DNS request results in a domain name, d, being resolved to an IP address, ip, at a particular time, t, forming a (d, ip, t) tuple. For each domain, $d \in \mathcal{D}_{f-tss}$, we compute two sets: (i) $RHIP(d)$, which is a set of all IPs that have mapped to domain d as recorded by the DNS Crawler (Section 4.2.3) within time window T, and (ii) $RHDN(ip)$, which is the set of domains that have historically been linked with the ip or $ip/24$ subnet in the $RHIP$ set within time window $T \pm \Delta$, where Δ is also a unit of time (typically one week). Next, we compute $\mathcal{D}_{rhip-rhdn}(d)$, which represents all the domains related to d at the network level, as discovered by the $RHIP$-$RHDN$ expansion. Now, for each domain $d' \in \mathcal{D}_{rhip-rhdn}(d)$, we check if the webpage $w_{d'}$ associated with it is a TSS webpage using the classifier module, Section 4.2.4. If it is true, we add d' to an amplification set, $\mathcal{D}'_{f-tss}(d)$, associated with d. The cardinality of the eventual amplification set gives us the amplification factor, $\mathcal{A}(d)$. Finally, we define the expanded set of TSS domains, $\mathcal{E}_{f-tss}$, as the union of all amplification sets. Combining the initial set of domains, $\mathcal{D}_{f-tss}$, with the expanded set, $\mathcal{E}_{f-tss}$, gives us the final set of fake-technical support domains $\mathcal{F}_{f-tss}$.

The data pertaining to historic DNS resolutions comes from the ActiveDNS Project [3], while the webpages associated with the new domains are obtained by the active HTML crawler module (Section 4.2.3) and, when required, the Internet archive [25]. The final technical support domain set is processed further for analysis.

4.2.6 Clustering Module

The purpose of the clustering module is to identify different TSS campaigns. For example, one campaign may offer technical support for Microsoft whereas another one may target Apple users. We identify the campaigns by finding clusters of related domain names associated with abuse in a given time period or epoch t. Once we have the final set of TSS domains, a two step hierarchical clustering process is used. In the first level, referred to as Network CLustering (NCL), we cluster together domain names based on the network infrastructure properties. In the second level, referred to as Application CLustering (ACL), we further separate the network level clusters based on the application level web content associated with the domains in them. This process allows us to produce high quality clusters that can then be labeled with campaign tags.

In order to execute these two different clustering steps, we employ the most common statistical features from the areas of DNS [54] and HTML [87, 93] modeling to build our feature vector. This feature vector embeds network information about not just the final landing domain d, but also of all the domains supporting d, based on the redirection path to d. The vector also captures the agility of the domains: if d resolved to multiple different IPs over time, this information would be present. We use Singular Value Decomposition (SVD) [100] to reduce the dimensionality of the sparse feature matrix, and the network clustering module then uses the X-Means clustering algorithm [83] to cluster domains having similar network-level properties. To further refine the clusters, we use features extracted from the full HTML source of the web pages associated with domains in $\mathcal{F}_{f-tss}$. We compute TF-IDF statistical vector on the bag of words on each cluster c [87]. Since the matrix is expected to be quite sparse, the application cluster submodule performs dimensionality reduction using SVD, like in NCL. Once we have the reduced application based feature vectors representing corresponding domains, this module too uses the X-Means clustering

algorithm to cluster domains hosting similar content.

Campaign Labels: This submodule is used to label clusters with keywords that are representative of a campaign's theme. Let C be a cluster produced after NCL and ACL, and let D_C be the set of domains in the cluster. For each domain $d \in D_C$, we create a set $U(d, T)$ that consists of all the parts of the domain name d except the effective top level domain (eTLD) and all parts of the corresponding webpage title T, e.g. U('abc.exampledomain.com', 'title') = {abc, exampledomain, title}. Next, we compute the set of words $W(U(d))$ using the Viterbi algorithm [65]. Therefore, W(U('abc.exampledomain.com', 'title')) = {example, domain, title} since 'abc' is not a valid English word or W(U('virusinfection0x225.site', 'System Shutdown Call 877-563-1632')) = {virus, infection, system, shutdown, call}. Using W, we increment the frequency counter for the word 'example', 'domain' and 'title' in a cluster specific dictionary. In this manner, after iterating over all domains in the cluster, we get a keyword to frequency mapping from which we pick the top most frequent word(s) to attribute to the cluster. Identifying campaigns this way allows us to study properties related to the campaign more readily.

4.3 Results

We built and deployed the system described in Section 4.2 to collect and analyze SR and AD domains for TSS. Although the system continues to be in operation, the results discussed in this section are based on data that was collected over a total period of 8 months in two distinct time windows, April 1 to August 31, 2016 initially, and again between Jan 1 - Mar 31, 2017, to study the long running nature of TSS.

Infrastructure Setup: We deploy two distinct nodes on a university network where the SEC and ACM modules for data collection run. One is a desktop class machine with 16GB RAM, a 3.1 GHz quad-core Intel Core i5 processor that runs Mac OS X 10.11. This node simultaneously runs the same data collection code on

a virtual machine with Windows Vista guest OS. The other node is a server class machine with 32GB RAM, 8 Intel Xeon quad core processors that runs the Debian 3.2.68 OS. We set the *User Agent* (UA) to be a version of Chrome, Internet Explorer and the Firefox browser respectively, covering the most commonly used browsers. The *Referer* field is set based on the search engine to which the process thread is attached. We clear the cookie field every time we query a search engine or make a request to an AD/SR URI. The IP addresses of the nodes are static and assigned from the university subnet. Previous studies [80] have shown that it is more effective to perform such threat data collection from university networks rather than from a public cloud infrastructure. We made similar observations from an experiment we conducted and chose the university network for our work. To make sure that none of the search engine operators throttle our crawlers, we rate limit the number of queries sent each day to a particular search engine.

We crawled 5 search engines for both ADs and SRs, which include Google.com, Bing.com, Yahoo.com, Goentry.com and search.1and1.com. The first three are popular search engines used daily by users while goentry was chosen because it has been linked with browser hijacking and serving unwanted ADs [37, 19]. The last search engine was added to the list after we encountered regular references/links to it among goentry ADs. Each day, the SEC module automatically sends 2,600 different queries, as discussed in Section 4.2.1 for technical support-related terms (e.g. microsoft tech support) to the various search engines. It stores the AD and SR URIs returned. We consider the top 100 SR URIs (unless there are fewer) while recording all the AD URIs displayed for each query.

4.3.1 Dataset Summary

In total we collected 14,346 distinct AD URIs and 109,657 distinct SR URIs. Table 8 presents the breakdown of all the search listings into the different categories. The AD

*Table 8: Categorization of Search Results. *Includes FakeCall, FakeBSOD, TechBrolo etc.*

	Advertisments (AD)				Search Results (SR)				AD+SR	
	URIs		Domains		URIs		Domains		Domains	
	#	%	#	%	#	%	#	%	#	%
TSS	**10,299**	**71.79**	**2,132**	**43.04**	**59,500**	**54.26**	**3,583**	**17.51**	**5,134**	**22.13**
Aggressive*	7,423	51.74	1,224	24.71	45,567	41.55	2,281	11.15	3,166	13.65
Passive	2,876	20.05	908	18.33	13,933	12.71	1,302	6.36	1,968	8.48
Non-TSS	**4,047**	**28.21**	**2,822**	**56.96**	**50,157**	**45.74**	**16,880**	**82.49**	**18,061**	**77.87**
Legitimate	1,892	13.19	1,442	29.10	3,726	3.39	3,499	17.09	3,790	16.34
Blogs/Forums	0	0.00	0	0.00	10,012	9.13	3,001	14.67	3,001	12.94
Complaint Websites	0	0.00	0	0.00	9,998	9.12	202	0.99	202	0.87
News	0	0.00	0	0.00	12,113	11.05	1,208	5.90	1,208	5.21
Uncategorized	2,155	15.02	1,380	27.86	14,308	13.05	8,970	43.84	9,860	42.51
Total	**14,346**	**100.00**	**4,954**	**100.00**	**109,657**	**100.00**	**20,463**	**100.00**	**23,195**	**100.00**

URIs mapped to 4,954 unique Fully Qualified Domain Names (FQDNs), while the SR URIs mapped to 20,463 unique FQDNs. Among the AD URIs, 10,299 (71.79%) were observed as leading to TSS websites. This is a significant portion and shows that ADs related to technical support queries are dominated by those that lead to real scams. It also means that the technical support scammers are actively bidding in the AD ecosystem to flood the AD networks with rogue technical support ADs, especially in response to technical support queries. Such prevalence of TSS ADs is the reason why Bing announced a blanket ban of online tech support ADs on its platform [8] in mid-May, 2016. The TSS AD URIs mapped to 2132 FQDNs. Among the TSS AD URIs and corresponding FQDNs, we found the presence of both aggressive and passive websites. More than two thirds of the URIs were seen to lead to aggressive websites. The ratio between aggressive and passive websites was closer to 4:3 when considering just the TSS AD FQDNs. Past research has only investigated aggressive TSS websites, but our results show that passive websites are also a serious problem.

We did observe legitimate technical support service AD URIs and FQDNs. These comprised about 13.19% of all AD URIs and 29.10% of all AD FQDNs. There were no ADs that pointed to blogs/forums, complaint websites and news sites. About 15% of the AD URIs remained uncategorized: however, it is worth mentioning that on manual inspection, one set of the URIs/domains seen in the uncategorized bucket

led to other shady (and perhaps temporary) search portals such as govtsearches.com, finecomb.com, us.when.com and many more. These search portals show more ADs and SRs in response to the original search query. This pattern of creating on-the-go search portals and linking them to each other via ADs to form a nexus is intriguing and worthy of exploration in itself. We leave this for future work.

Among the SR URIs, 59,500 (54.26%) were observed leading to TSS websites. The URIs mapped to 3,583 (17.51%) FQDNs. Among the TSS SR URIs, we again found the presence of those leading to both aggressive and passive TSS varieties. The sheer number of such URIs is surprising as, unlike ADs, it is harder to manipulate popular search engine algorithms to make rogue websites appear in search results. However, as we discuss later, we observe that using black hat SEO techniques, TSS actors are able to trick the search engine ranking algorithms. Compared to ADs, we found that almost 76% TSS SR URIs lead to aggressive TSS websites while the remaining lead to passive TSS websites, again pointing to the prevalence of the common tactic of scare and sell [70]. Although TSS SR URIs were frequently seen interspersed in search results, SR URIs also consisted of non-TSS ones. Among these we observed 3.39% legitimate technical support service URIs, 9.13% blog/forum URIs, 9.12% URIs linked to complaint websites and 11.05% URIs pointing to news articles (mostly on technical support scams). The remaining 13.05% URIs were uncategorized.

We also report aggregate statistics for FQDNs after combining ADs and SRs data. We see that in total there were 5134 TSS FQDNs found, with URIs corresponding to 3166 FQDNs leading to aggressive websites and 1968 leading to passive websites. These together comprise of about 22.1% of the total number, 23,195 FQDNs retrieved from the entire dataset. One interesting observation is that majority of the FQDNs seen in ADs were not seen in the SRs and vice versa, with only a small amount of overlap in the TSS AD FQDNs and TSS SR FQDNs, consisting of 581 FQDNs. It suggests that the resources deployed for TSS ADs are different from those appearing

in TSS SRs.

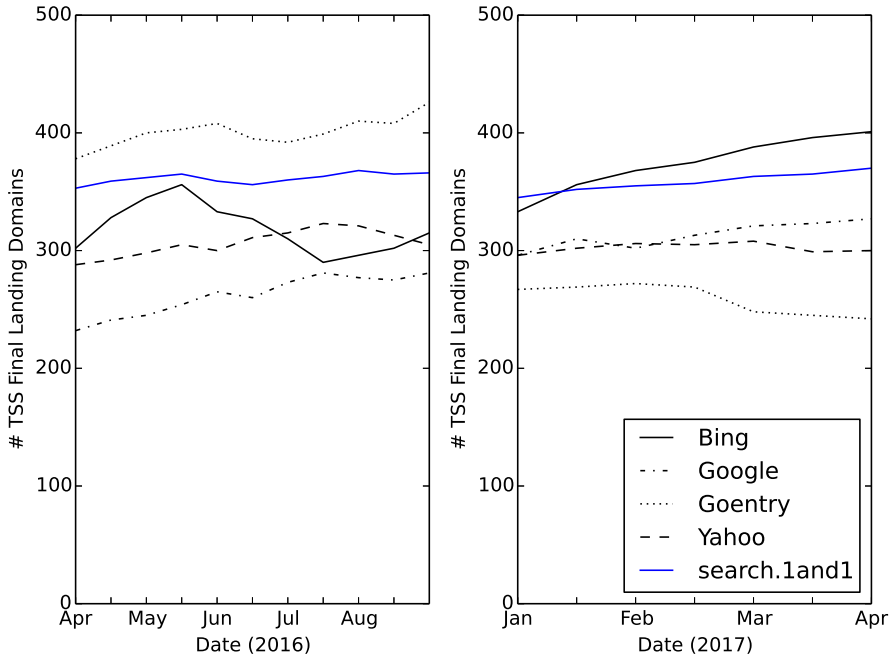


Figure 17: Bi-weekly trend of the number of final landing TSS domains found classified based on the search engine of origination for the two time periods of data collection.

Support and Final-landing TSS domains: The purpose of support domains is to conduct black hat SEO and redirect victims to TSS domains but not host TSS content directly. We found 38.3% of the TSS search listing URIs did not redirect to a domain different from the one in the initial URI, while the remaining 61.7% redirected to a domain different from the one in the initial URI. There were an additional, 2,435 *support* domains found. Moreover, one might expect the use of popular URL shortening services such as bit.ly or goo.gl for redirections and obfuscation, but this was rarely the case, which we found surprising.

When a TSS URI appearing in the search listings is clicked, it leads to the webpage that lures the victim into the technical support scam. This webpage could be hosted on the same domain as the domain of the URI, or on a different domain. We refer to this final domain name associated with the technical support scam webpage as the final landing TSS domain. Furthermore, it is possible that the path from the initial

SR/AD URI to the final landing webpage consists of other intermediate domains, which are mainly used for the purpose of redirecting the victim's browser. This is discussed in Section 4.2.3. Figure 17 plots the number of final-landing TSS domains discovered by our system over time across the various search engines. A bi-weekly trend shows that, across all search engines, we are able to consistently find hundreds of final-landing TSS domains and webpages. Bing, Google, Goentry, Yahoo and search.1and1.com, all act as origination points to technical support scam webpages. This suggests that these specialized scammers are casting a wide net. Starting mid-May 2016, we see a sudden dip in the number of TSS domains found on Bing. We suspect that this is most likely correlated to Bing's blanket ban on technical support advertisements [8, 7]. However, as we can see, activity, contributing mainly to SR based TSS, picked up again during July, 2016, continuing an upward trend in Jan to Mar 2017.

Goentry, which was a major source of technical support ADs leading to final landing TSS domains during our initial period of data collection saw a significant dip during the second time window. We suspect this may be due to our data collection infrastructure being detected or law enforcement actions against technical support scammers in India [24, 23], which is where the website is registered. In total we were able to discover 1,626 unique AD originated final landing TSS FQDNs, and 2,682 unique SR originated final landing TSS FQDNs. Together, we were able to account for 3,996 unique final landing TSS FQDNs that mapped to 3,878 unique final landing TSS TLD+1 domain names.

4.3.2 Search Phrases Popularity and SR Rankings

Since we use search queries to retrieve SRs and ADs, one question is the popularity of search phrases used in these queries which can serve as an indicator of how frequently they are used to find tech support related websites. We use popularity level derived

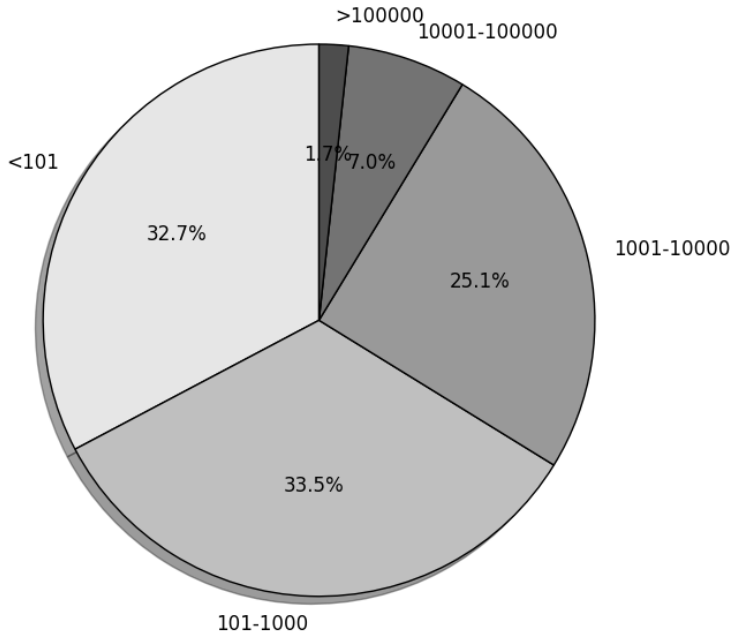


Figure 18: Fraction of technical support phrases with the corresponding average global monthly searches on Google during the months of threat data collection. Dataset consists of both popular and not so popular search phrases.

from Google's keyword planner tool [27] that is offered as part of its AdWords program. The popularity of a search phrase is measured in terms of the average number of global monthly searches for the phrase during the time period of data collection. Figure 18 shows the distribution of technical support search phrases based on their popularity. We can see that out of the 2600 phrases associated with TSS, about one third (32.7%) were of very low popularity, e.g. *'kaspersky phone support'* with less than 100 average global monthly searches, one third (33.5%) were of low popularity, e.g. *'norton antivirus technical support'* with 101-1,000 hits per month on average, while there were 25.1% phrases that had medium levels of popularity, e.g. *'hp tech support phone number'* with 1,001-10,000 average hits. At the higher end, 7% of the technical support phrases had moderately high levels of popularity, e.g. *'dell tech support', 'microsoft support number'* with 10,001-100,000 hits per month on average, and 1.7% of the technical support search phrases were highly searched for, e.g.

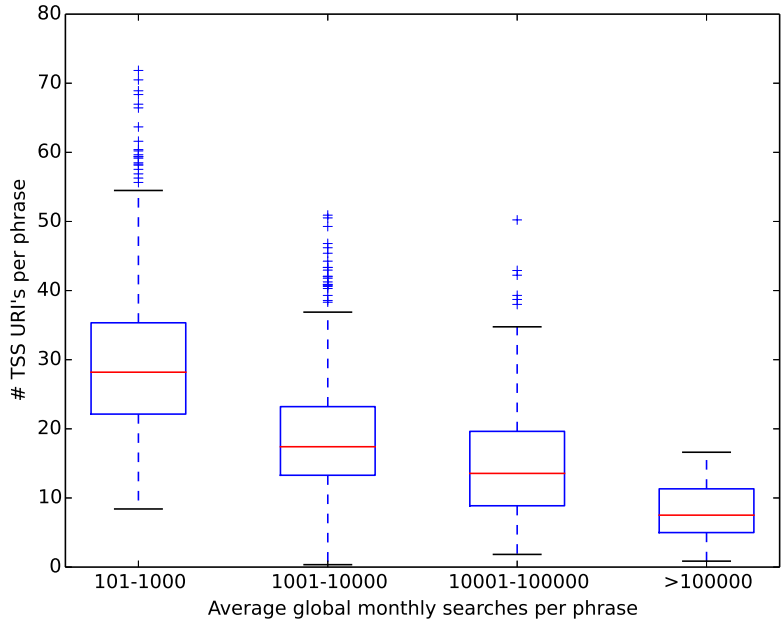


Figure 19: Relationship between popularity of a search phrase and the TSS URI pollution levels in the search listings. URI counts include AD and SR URI's as seen on Google. Phrases with popularity less than 100 average hits per month ignored.

'lenovo support' with greater than 100,000 hits per month globally. As we can see, we have a fairly even distribution of technical support search terms with varying levels of popularity ranging from low to high (in relative terms).

One may expect that less popular search terms are prone to manipulation in the context of both ADs and SRs, while more popular ones are harder to manipulate due to competition, making it more difficult for the technical support scammers to promote their websites via bidding (in the case of ADs) or SEO (in the case of SRs). To validate this, we measure the number of total TSS URIs found per search phrase (referred to as *pollution* level), as a function of the popularity of the phrase. Since the popularity levels of phrases are gathered from Google, we only consider the TSS URIs (both AD and SR as seen on Google) to make a fair assessment. Figure 19 depicts a box plot that captures the pollution levels for all search phrases grouped by the popularity levels except the ones with very low popularity. By comparing the median

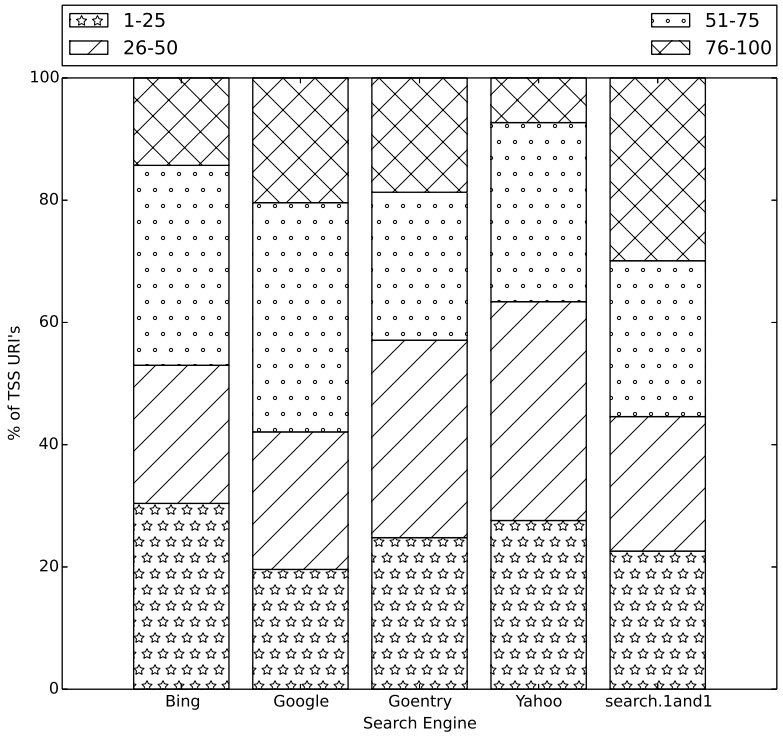


Figure 20: Distribution of TSS SR URIs based on the position in search listings for different search engines.

number of TSS URIs (depicted by the red line(s)) from different popularity bands, we witness that as the popularity level of a search term increases, the pollution level (i.e. the absolute number of TSS URIs), decreases. We can make several additional observations: (i) there is definite pollution irrespective of the popularity level: in other words, more than a single TSS URI appeared in almost all of the technical support search queries we considered, as can be seen from the floor of the first quartile in every band; (ii) while many (∼50%) low popularity search terms (e.g. those with 101-1000 hits per month) yielded 28 or more TSS URIs, there were outliers even among the high popularity search terms that accounted for the same or even more number of TSS URIs; and lastly, (iii) the range in the number of TSS URIs discovered per query varied more widely in the case of low popularity terms as compared to higher popularity terms. Overall, these results indicate that TSS scammers are intent on pushing their target websites among (i) high-impact results, in spite of the challenges

in doing so, while (ii) simultaneously picking low hanging fruits by widely spreading their websites among the search listings associated with less popular technical support search queries.

To effectively target victims, it is not merely enough to make TSS URIs appear among the search results. It is also important to make them appear high in the search rankings. To measure this, we show the distribution of TSS SR URIs based on their ranking/position among the search results for different search engines. We use four brackets to classify the TSS SR URIs based on its actual position: 1-25 position (high rank), 26-50 position, 51-75 position and 76-100 position (low rank). If the same URI appears in multiple search positions, for example on different days, we pick and associate the higher of the positions with the URI. We do this to reflect the worst-case impact of a TSS SR URI. Thus, each unique URI is eventually counted only once.

Figure 20 summarizes our findings. We see that all 5 search engines return TSS URIs that are crowding out legitimate technical support websites by appearing high in the rankings. For a more fine grained analysis of the rankings and its potential impact, out of the top 25 positions, we measured the fraction of TSS SR URIs appearing in the top three as well as the top ten positions. We found that Bing had the highest percentage, 8% of TSS SR URIs appearing among the top three positions and 17% TSS SR URIs appearing in a top ten spot. Even the other search engines had their top three and top ten search positions polluted regularly by TSS URIs. This makes it hard to trust a high ranking URI as legitimate. Bing had the highest percentage (30.4%) among all its TSS URIs appearing in the top 25 search results, followed by Yahoo (27.6%), Goentry (24.8%), search.1and1 (22.6%) and Google (19.6%). Note that here we are not comparing the absolute number of TSS URIs between the search engines. TSS URIs are seen distributed across all position bands, again pointing to the pervasive nature of the TSS pollution problem.

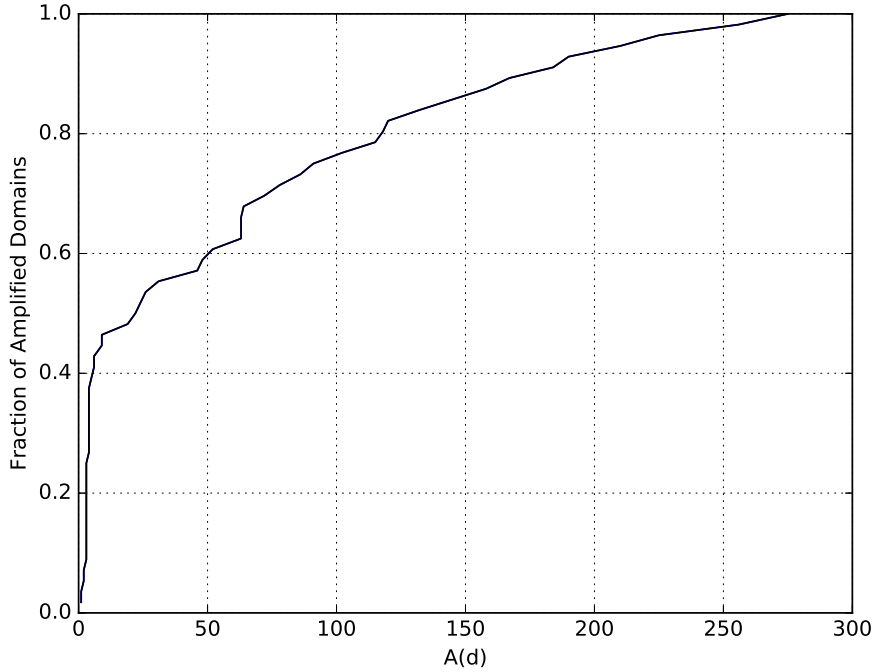


Figure 21: CDF of the network amplification factor, $\mathcal{A}$, of final landing TSS domains discovered using search listings.

4.3.3 Network Amplification Efficacy

The network-level amplification approach did pose a number of challenges. The first challenge lies in the fact that sometimes technical support websites are hosted on public cloud infrastructure. Thus, the set $\mathcal{D}_{rhip-rhdn}$for such domains can yield an overwhelming number of domains to process for the TSS webpage classifier. We avoid this by excluding *rhip-rhdn* sets, $\mathcal{D}_{rhip-rhdn}(d)$, having size greater than a reasonable operator specified threshold, λ. The other challenge lies in the fact that sometimes the webpages associated with the *rhip-rhdn* domains, $w_{d'}$, are not retrievable. This could be because the webpage is parked, taken down or expired. Further, even the Internet archive may not have snapshots of the webpage associated with the domain in the desired time window. In such cases, we are forced to exclude the domain from further consideration even when there is evidence of it being linked to technical support scams, e.g. based on the domain name itself.

Using these heuristics, and dropping any domains having amplification factor $\mathcal{A}(d) < 1$, we are conservatively left with only 2,623 domains in the $\mathcal{D}_{f-tss}$ set that contributed to the *rhip-rhdn* expansion set, $\mathcal{E}_{f-tss}$. Figure 21 plots the cumulative distribution of the amplification factor of these domains. As we can see, around 60% domains had $\mathcal{A}(d) \leq 50$ while the remaining 40% domains had $\mathcal{A}(d) > 50$, with the maximum $\mathcal{A}(d)$ value equal to 275. Note that there could be overlap between the amplification sets, $\mathcal{D}'_{f-tss}(d)$, for different d's. Also worth noting is the fact that having a low amplification value does not necessarily mean that there are no other TSS domains on the subnet as it could be that some of DNS records associated with domains on the network were not previously recorded/seen by the deployed sensors. With ISP scale DNS records, the amplification values can potentially be much greater. In all, the total number of unique FQDNs hosting TSS content, $|\mathcal{F}_{f-tss}| = 9{,}221$, with 3,996 TSS FQDNs coming from the final landing websites in search listings and 5,225 additional TSS FQDNs discovered as a result of network-level amplification. These 9,221 FQDNs mapped to 8,104 TLD+1 domains. Thus, even though amplification is non-uniform, it helps in discovering domains that may not be visible by search listings alone.

The network amplification process allowed us to identify 840 passive-type TSS domains co-located with one or more aggressive TSS domains. This indicates that some of the passive scams are operated by the same scammers who operate the aggressive ones. This is likely part of a diversification strategy where, depending upon the method of retrieving users, scammers can show different types of pages: e.g. aggressive ones for those involved in "malvertising" redirections and passive ones for those that are already in the market for technical support services.

4.3.4 Domain Infrastructure Analysis

In this section, we analyze all the domain names associated with technical support scams discovered by our system. This includes the final landing domains that actually host TSS content as well as support domains, whose purpose is to participate in black hat SEO or serve as the redirection infrastructure.

Most abused TLDs: First, we analyze the final landing TSS domain names. Table 9 shows the most abused TLDs in this category. The *.com* TLD appeared in 25.56% final landing TSS domain names, making it the most abused TLD. Next, 16.21% domain names had *.xyz* as the TLD, making it the second most abused TLD. *.info*, *.online* and *.us* each had greater than 6% domain names registered to them completing the top five in this category. Other popular gTLDs included *.website, .site, .tech, .support*, while the ccTLDs included *.in, .tk, .co* and *.tf*. Among the *support* domains, the top three most popular TLDs were *.xyz, .win* and *.space*. Although *.xyz* was once again very popular like in the case of the final landing TSS domains, both *.win* and *.space* were exclusive to this category. We also compared the TLDs associated with the final landing TSS domain names with those discovered by ROBOVIC, the system developed by Miramirkhani et al. [80]. For an overlapping data collection period between January to March of 2017. We found that 4 out of the top 10 TLDs associated with TSS domains served by abusing domain-parking and ad-based URL shortening services were different from those discovered in our dataset. The TLDs that were rarely visible in our dataset included *.club*, *.pw*, *.trade* and *.top*. Thus, there are differences with respect to the preference of domain name registration between these two different tactics.

Domains Lifetimes: Next, we look at the lifespan of final landing and support domains. The lifetime of a final landing TSS domain is derived by computing the difference between the earliest and most recent date that the domain was seen hosting TSS content. This computation is based on data from our crawler and the Internet

Table 9: Most abused top-level domains (TLDs) used in final-landing TSS websites.

TLD	%
com	25.56
xyz	16.21
info	7.62
online	6.78
us	6.34
net	5.91
org	4.86
in	4.44
website	4.10
site	3.69
tk	2.03
tech	2.12
co	1.89
tf	1.67
support	1.44
others	5.34
Total	100

archive. The lifetime of a support domain is derived based on earliest and the most recent date that the domain was seen redirecting to a final-landing TSS domain. Figure 22 plots the lifetimes of these two categories of domains with the final landing domains split up into the passive and aggressive types. Final landing TSS domains of the aggressive type had a median lifetime of ∼9 days with close to 40% domains having a lifetime between 10-100 days, and the remaining ∼10% domains having a lifetime greater than a 100 days. In comparison, final landing TSS domains of the passive type had a much longer median lifetime of ∼100 days. Some of the domains in this category had a lifetime of over 300 days. Clearly, passive TSS domains outlast those of the aggressive type. The reason for this could be attributed to the nature of these domains, with the aggressive domains being clear candidates for reporting/take-down and the passive ones getting the benefit of doubt (as they tend to appear legitimate and conduct the fraud mainly via the phone channel). Irrespective of the reason, it suggests that passive TSS websites have the potential to do harm for

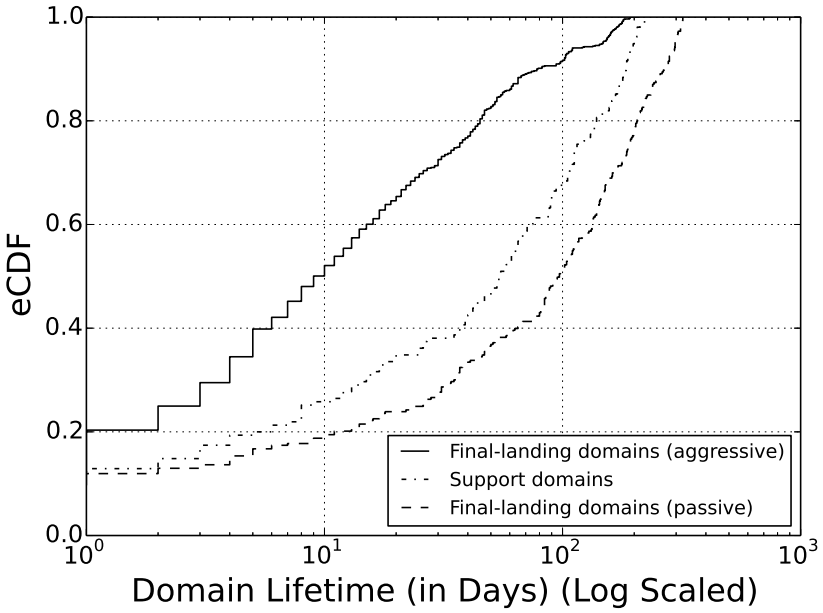


Figure 22: Lifetime of different types of TSS domains

long time periods. In comparison, support domains had a median lifetime of ∼60 days, with ∼33% domains having a lifetime greater than 100 days. Generally, this is a longer lifetime relative to final landing TSS domains of the aggressive type. It indicates that the domains that are used for the sole purpose of black hat SEO or redirection are relatively stable and reusable (due to the long-lived nature), helping their cause to redirect to final landing TSS domains when desired and yet remain unnoticed. As we discuss later, in addition to blacklisting the final landing domains, take down/blacklisting of these *support* domains would lead to a more effective defense in breaking parts of the TSS abuse infrastructure.

Overlap with Blacklists: Using domains and phone numbers from a large number of public blacklists (PBL) [32, 1, 21, 50, 31, 43, 42, 26, 38, 22, 2, 29], we verify if and when a TSS resource appeared in any of the PBLs. We collected data from these lists beginning Jan 2014 up until April 2017, encompassing the AD/SR data collection period, which allows us to make fair comparisons.

Table 10: Overlap between final landing TSS domains with popular public blacklists. $^{+}$includes Malware Domains List, sans, Spamhaus, itmate, sagadc, hphosts, abuse.ch and Malc0de DB.

Blacklist Name	Coverage (in %)		Type
	FQDN	TLD+1	
Malwarebytes TSS List	18.1%	n/a	Telephony BL
Google Safe Browsing	9.6%	5.2%	Traditional DBL
800notes.com	14.2%	n/a	Telephony BL
VirusTotal	22.6%	10.8%	Traditional DBL
Others^{+}	5.3%	3.4%	Traditional DBL
Cumulative	26.8%	12.5%	

We start with 800notes.com, which is a crowdsourced directory of unknown callers. It consists of complaints by users who post about telephony scams, not just technical support scams. We extracted domain names appearing in the complaints. We find that only 14.2% of final landing TSS FQDNs were reported in the complaints.

Next, we look at a more exclusive TSS blacklist released by Malwarebytes. These public blacklists are specific to TSS and are regularly updated. According to their website, they use both crowdsourced and internal investigations to generate the list. Over time there were 4,949 unique FQDNs listed on the list. We found that 18.1% of the final landing TSS FQDNs identified by our system were also listed in these lists.

Next, we queried the domains against the Google Safe Browsing list using their API. We found that 9.6% final landing TSS FQDNs and 5.2% second-level domains (TLD+1) from those identified by our system as fake technical support were also listed in Google's system and were all labeled as "Social Engineering." Since Google does not have a public list of abusive phone numbers, we leave the corresponding field blank. Lastly, we checked PBLs that typically include botnet C&C domains, malware sites and other unsafe domains serving malicious content. These include all the lists mentioned before except 800notes, Malwarebytes TSS list, and Google safe browsing list. We find that together these cover just 5.3% FQDNs and 3.4% TLDs from our list.

Next, we check the domains against VirusTotal which is comprised of feeds from multiple AV engines. We found that 22.6% final landing TSS FQDNs and 10.8% TLD+1s listed on it. While this list gave the greatest coverage in TSS domain name blacklisting, we still found significant scope for improvement in terms of coverage. Moreover, it is still lower (in relative terms) as compared to the findings by Miramirkhani et al. [80] where close to 64% of their TSS domain set was listed on VirusTotal. We suspect that this is in part due to some of the passive TSS domains which largely go undetected. Overall, these results are not very surprising since these are traditional blacklists whose intelligence is targeted towards other types of abusive domains, such as, botnet domains. A similar outcome in terms of the efficacy of these lists has been reported in an SMS-spam domain abuse [93].

Cumulatively, these lists cover only 26.8% FQDNs, that were found to be involved in TSS by our system. Moreover, out of the 26.8% blacklisted FQDNs, 8.2% were already present in one of the lists when our system detected them, while the remaining 18.6% were detected by our system $\sim$26 days in advance, on average. Moreover, when we cross-listed the *support* domains against these lists, we found that $<$1% of those were present in any of these lists. This reinforces the point made in Section 4.3.4 regarding blacklisting *support* domains for effective defense against TSS. Table 10 summarizes these findings.

4.3.5 Campaigns

The Clustering module (Section 4.2.6) produces clusters consisting of final landing domains that share similar network and application features. Table **??** lists some of the major campaigns attributed by our system and the resources associated with them. First, although TSS are notoriously synonymous with Microsoft and its products, we found that many other brands are also targets of TSS campaigns. These

brands include Apple, Amazon, Google and Facebook among others. Microsoft, however, remains on top of the most abused brands with 4 out of the top 5 TSS campaigns targeting Microsoft and its products. Second, we observed that TSS campaigns tend to advertise services targeted at particular brands and its line of products. For example, certain TSS campaigns advertise services only for Gmail accounts or Norton Antivirus or Firefox browser or the Windows OS. The outcome from the victim's perspective can vary depending on the product: examples include credential theft, genuine product key phishing, browser compromise and remote hijacking of the OS in the aforementioned cases respectively. This behavior is likely because the call center agents are trained to specialize in technical aspects associated with a particular type of product/service which could be a device (e.g. kindle), software (e.g. browser) or OS (e.g. Windows Vista) rather than generic technical support. Such a brand based view can be used to alert companies about campaigns targeting them so that they in turn can take appropriate action in stemming the campaign or alert their users about it.

The identified campaigns allow us to study the relationship between domains and the phone numbers advertised by them. We find that the churn rate in phone numbers is comparable to the churn rate of domain names for certain campaigns while there also exist campaigns where the churn rate in phone numbers is very low as compared to the domains names. Evidence of both cases is present in Table **??**. The first, third and fourth campaigns listed in the table represent a `N-N` relationship between domain names and phone numbers i.e. each final-landing domain associated with the campaigns is likely to advertise a different phone number. However due to the routing mechanism of the toll-free numbers, the calls to them may end up in the same call center. On the contrary, the second to last campaign depicts a `N-2` relationship between domain names and phone numbers with 42 final-landing domains sharing just 2 toll-free numbers. By analyzing the clusters produced, we find that the presence

of support domains is not ubiquitous. Only certain campaigns tend to make use of support domains. Furthermore, these campaigns are associated with SEO behavior and tend to be of the aggressive type.

Finally, to evaluate the efficacy of the clustering module, we inspect randomly selected large and small clusters and verify that the domains clustered together are indeed part of a TSS campaign. We are able to use ground truth data from Malwarebytes and human intelligence where required to evaluate the outcome of the clusters and conclude that the hierarchical clustering methodology works well.

In total, 368 clusters were produced after both network and application level hierarchical clustering. Next, we present case studies of two campaigns to highlight TSS tactics.

Table 11: Selected large campaigns, as identified by the clustering module.

Final landing domains	Support domains	IPs	Phone Numbers	Clustering Label(s)	Sample Domains
662	452	216	521	microsoft virus windows	call-po-1-877-884-6922.xzz0082-global-wind0ws.website, virusinfection0x225.site
232	0	38	112	amazon kindle phone	kindlesupport.xyz
199	172	112	199	microsoft technician vista windows	talktoyour-technician.xyz
91	43	134	46	error microsoft threat	error-go-pack-akdam-0x00009873.website, suspiciousactivitydetectedpleasecallon18778489010tollfree.*.lacosta.cf
82	0	21	43	key office product	officesetupzone.xyz
76	0	36	38	antivirus norton	nortonsetup.online
75	0	18	28	browser firefox	firefoxtechnicalsupport.com
68	0	23	36	gmail login	gmailsupportphonenumber.org
55	0	41	51	chrome google	chromesupportphonenumber.com
48	22	42	47	apple risk	apple-at-risk.com, apple-atrisk.com
42	0	10	2	code error network	networkservicespaused.site, 04cve76nterrorcode.site
36	0	12	15	customer facebook service	facebooksupportphonenumber.com

4.4 Case Studies

To gain deeper insights into TSS abuse infrastructure, we discuss two specific case studies. The first one illustrates the use of support domains for black hat SEO and the second one demonstrates the use of browser hijacking to serve TSS ads.

4.4.1 Black Hat SEO TSS Campaign

In this case study, we analyze the largest TSS campaign from Table 11 to highlight the technique used to promote the TSS websites and the infrastructure used to grow and sustain the campaign over time. The campaign primarily targeted `Bing.com` users. It consisted of 452 support domains, 662 final landing domains which mapped to 216 IPs over time and advertised 521 unique phone numbers. The campaign was first detected on 04/16/2016 and was active as recently as 03/30/2017. This is based on the first and last date on which the domains belonging to this campaign were identified by our system and added to the TSS dataset.

A search for "microsoft tech support", for instance, would yield a TSS support domain such as *zkhubm.win* among the SRs. Clicking on the SR would redirect the user's browser to a final landing TSS domain. The domain then uses aggressive scareware tactics to convince the victim about an error in their Windows machine. Then the victim is coerced into contacting the TSS call center. The social engineering and monetization would then take place over the phone channel, thus completing a typical TSS. Table 12 lists some of the support domains and the final landing domains to which they redirect.

SEO Technique: The support domains use black hat SEO techniques sometimes referred to as *spamdexing* to manipulate the SRs. Specifically, the support domains seen on the search page act as *doorway* pages to final landing TSS domains. However, they use cloaking techniques such as text stuffing and link stuffing, consisting of technical support related keywords and links, to hide their real intent from search

Table 12: Some of the ***support*** *and* ***final-landing*** *domains seen in the largest TSS campaign from our dataset.*

Support Domain	Final-landing Domain
zkhubm.win	err365.com, jo0dy-gmm-0003210.website
03d.gopaf.xyz	supportcorner.co, xoaodkfnm.tech, attorneylowerguide.xyz
gmzlpx.space	web-server-threat-warning0x58z2.us
utwwxs.win	call-po-1-877-884-6922.xzz0082-global-wind0ws.website, call-kl-1-877-884-6922.jo0dy-gmm-0003210.website
lgncbq.win	virusinfection0x225.site

engine crawlers and get promoted up the SR rankings. Figure 23 shows what a crawler/user would see when visiting a support domain if the Referer header does not indicate that the originating click happened on a search results page.

IP Infrastructure Insights: Figure 24 shows the spread of the IPv4 address space and how domains in this campaign map to it. We plot the fraction of support and final landing TSS domains as a function of the IP address space and make the following observations: (i) IP space used by support domains is quite different and decoupled from where final landing TSS domains are located, and ii) while the address space for fake technical support domains is fragmented, the entire set of support domains are concentrated in a single subnet, 185.38.184.0/24. IP to AS mapping for the subnet points to AS# 13213 under the name UK2NET-AS, GB. The ASN has country code listed as ME, Montenegro. The IP-Geo location data too points to an ISP in Montenegro, Budva. In contrast, IP's associated with final landing TSS domains pointed to different AS#'s 31815, Media Temple, Inc, AS# 13335, Cloudflare and AS# 26496 GO-DADDY-COM-LLC based on IP to AS mapping data. They were

Figure 23: Screenshots demonstrating Black-hat SEO behavior by the `support` *domain* `zkhubm.win`*. Left side of the figure shows a text-stuffed page when the domain is visited by a vanilla crawler. The right side shows the* `final-landing` *domain* `err365.com` *after redirection when the corresponding SR is clicked.*

geographically located in the US based on IP-Geo data. The fragmentation in the hosting infrastructure for the final landing TSS domains gives the technical support scammers a reliable way to spread their assets. The decoupling of the infrastructure between support domains and final landing TSS domains indicates that the technical support scammers are using the support domains as a "service" to offload the work of SEO. These support domains could well serve other types of scams and command a price for their specialization at a later time or in parallel. Finally, from a defense perspective, focusing takedown efforts on these intermediate domains will likely have a larger effect that the takedown of individual final TSS domains.

4.4.2 Hijacking the Browser to Serve TSS ADs: Goentry.com

Goentry.com has been linked with browser hijacking where malicious software changes the browser's settings without user permission to inject unwanted advertisements into the user's browser [37, 19]. We noticed Goentry.com serving TSS ADs during the initial stages of this research and decided to probe it further. We use this case study to provide insights into evolving tactics being used by TSS actors.

Website Content: The homepage of goentry.com is a simple page with a Goentry logo, a search bar and a tagline "Goentry protects you from Government/NSA Spying on

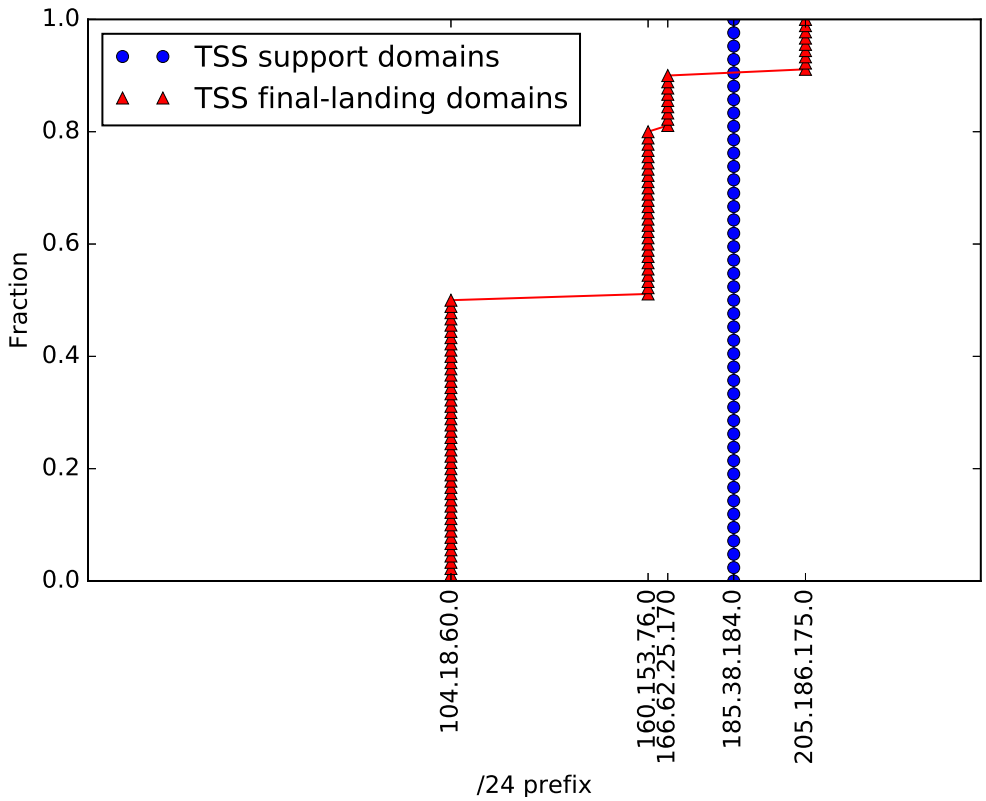


Figure 24: Fraction of Domains as a function of the IP address space.

your Searches." The output for a search query contains ADs on the top and right side of the results page, related search terms followed by search results. The website's root directory reveals content related to Goentry's SEO and website design services that includes their contact, toll-free number.

Based on the source, we observed the common use of the Universal Event Tracking (UET) [49] tags as trackers which allow the scammer to measure analytics such as the number of people that visited a specific page or a section of the website, amount of time they spent on the website etc. For example, the following code snippet corresponding to an AD seen on goentry.com, *gosearch770.xyz*, is tracked with UET tag id 54080586, and acts as a doorway page which redirects to fake technical support websites such as *error-error-error-2.xyz, critical-warning-message-2.xyz, portforyou.xyz* and many others, while monitoring the site analytics.

```
<div class="row_content">
  <div class="title">
    <a href="http://54080586.r.msn.com/?ld=d3S-92s04zd0_u=www.gosearch770.xyz2findex.php" target
        ="_blank">Com</a>
  </div>
  <div class="description">Browse Now For Com Online</div>
  <div class="link">
    <div class="link-left" style="float:left;"> gosearch770.xyz </div>
  </div>
</div>
```

Server-side Scripts: Our initial suspicion was that the search service is either using readily available APIs such as Google's Custom Search Engine to power their searches or it was running customized scripts. Using the source of the page, we found references to a server-side PHP script. Due to configuration errors from the side of the scammers, we were able to obtain parts of that script which revealed that the search-results page would react to the presence of certain keywords by adding tech-support ads to the returned page (e.g. when the users would search for the word "ice" the returned page would include ads about tech-support and the removal of a specific strain of ransomware called "the ICE Cyber Crime Center virus" [46]).

Domain Registration and IP: The website's registration records show that it was created on 01/22/2014 to an organization called Macrofix Technical Services Private Limited which is associated to the website `macrofix.com`. This website advertises techical support services and is known to be a scam [28].

4.5 Comparison with ROBOVIC

For the purpose of a direct comparison, we were able to obtain data from Miramirkhani et al. [80] for the period Jan-Mar 2017, which overlaps with the second time window of data collection in our work. Specifically, we received a list of 2,768 FQDNs discovered by their tool (2441 second-level domains), 882 toll-free phone numbers and 1,994 IP addresses. Upon intersecting these sets with our own data, we found 0/2,768 FQDNs and 0/2,441 second-level domains that were common. Moreover, in

terms of server and telephony infrastructure, we discovered that the two datasets had 92/1,994 common IP addresses of servers hosting TSS and 5/882 common toll-free phone numbers. We also discovered frequent use of "noindex" [10] meta tags in the HTML source of webpages associated with domains in Miramirkhani et. al. dataset which was noticeably missing from webpages in our dataset. Given this near-zero intersection of the two datasets, we argue that our approach is discovering TSS infrastructure that ROBOVIC is unable to find. Next to discovering aggressive tech support pages that ROBOVIC missed, a core contribution of our work is focusing on "passive" TSS which manifest mostly as organic search results. These pages are unlikely to be circulated over malvertising channels: a benign-looking tech support page is unlikely to capture the attention of users who were never searching for technical support in the first place. Since public blacklists are still unable to capture the vast majority of TSS (Section 4.3.4), our work complements the work of Miramirkhani et al. Specifically, by taking advantage of our system, blacklist curators would *double* the number of TSS domains, IP addresses, and phone numbers that could be added to their blacklists.

4.6 Summary

In this chapter, we analyzed Technical Support Scams (TSS) by focusing on two new sources of scams: organic search results and ads shown next to these results. Using carefully constructed search queries and network amplification techniques, we developed a system that was able to find thousands of active TSS. We identify the presence of long-lived support domains which shield the final scam domains from search engines and shed light on the SEO tactics of scammers. In addition to aggressive scams, our system allowed us to discover thousands of passive TSS pages which appear professional, and yet display phone numbers which lead to scammers. We showed that our system discovers thousands of TSS-related domains, IP addresses,

and phone numbers that are missed by prior work, and would therefore offer greater visibility into network infrastructure that is used to facilitation TSS.

CHAPTER V

CROSS-CHANNEL INTELLIGENCE SHARING

One of the main advantages that cross-channel abuse offers to the attacker is that it helps in evading the existing defenses that are deployed on the Internet and telephony channels respectively. The reason for this is that the defenses deployed in any one channel - Internet or telephony - rely on intelligence that is specific to only that channel, without taking into account attacks that could surface due to the convergence of these channels. For instance, telephony abuse, at first may not seem related to abuse of the domain and IP infrastructure on the Internet. However, as the cases discussed in the previous chapters have shown, they can indeed be related.

Cross-channel attacks abuse resources from both the Internet and telephony channels and it is possible that intelligence gathered from each channel can be further correlated to bolster defenses across the two channels. For instance, as work in Chapter 3 showed, intelligence using SMS-spam data, collected from the telephony channel, can be used to identify malicious domain and IP infrastructure on the Internet channel. Similarly, work in Chapter 4, showed that intelligence about TSS domains, collected on the Internet channel, can help identify abuse end-point such as phone numbers on the telephony side. In this chapter, we seek to delve deeper into cross-channel intelligence sharing, where intelligence from one channel can be leveraged to enhance the defenses on the other channel. In doing so, we explore if cross-channel intelligence sharing can augment existing defenses such as blacklists, thereby increasing their `coverage` and `timeliness` while minimizing the resources required to translate useful signals into meaningful action and defense.

Specifically, we show the following: (i) currently effective cross-channel intelligence sharing is not occuring which results in limited coverage of abuse related domain names and phone numbers in traditional blacklists and blocklists respectively, and (ii) timely sharing of intelligence between the two channels could be beneficial by reducing the threat exposure time to abusive cross-channel domains and phone numbers. We show that cross-channel intelligence sharing is likely to benefit all parties involved, such as the telcos, Internet service providers, regulatory and law enforcement authorities, brands, and most importantly the consumers.

5.1 Sharing of Intelligence: Telephony to Internet

The telephony channel can be a useful and untapped source of intelligence in detecting abuse infrastructure on the Internet channel. SMS-data that is collected on the telephony channel can be used for this purpose. In this section, we elaborate on the benefit that telephony abuse intelligence can offer to the Internet channel.

5.1.1 Telephony abuse and its relation to Internet blacklists

To provide evidence of the lack of intelligence sharing from telephony datasets to the online channel, we take domains from public blacklists (PBL), namely 'Malware Domains List' [31], 'sans' [43], 'Spamhaus Blacklist' [42], 'itmate' [26], 'sagadc' [38], 'hphosts' [22], 'abuse.ch' [2] and 'Malc0de' Database [30], and we verify if and when an SMS-spam domain appeared in any of the PBLs. These PBLs typically include phishing domains, botnet domains, malware sites and other unsafe domains serving malicious content. Given that the cross-channel domains are alive for a long time, as shown in Chapter 3, and that the cross-channel spamming phenomenon is relatively new, it was not clear whether the traditional blacklists are keeping pace with SMS-spam domains. Indeed, our finding shows that SMS-spam abuse is practically unknown to the PBLs. In total, we had only 177 out of the 17,528, a mere 1%, fully qualified domain names (FQDNs) listed in PBLs. Out of this, 170 domains appeared

Table 13: Coverage in telephony to Internet intelligence sharing scenario. $^{+}$includes Malware Domains List, sans, Spamhaus, itmate, sagadc, hphosts, abuse.ch and Malc0de DB.

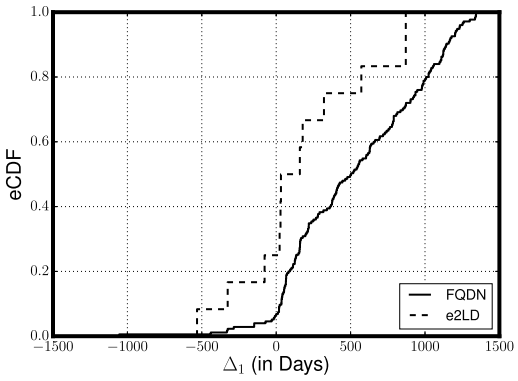
Blacklist Name	Coverage (in %)	
	FQDN	TLD+1
PBLs^{+}	1%	0.08%

in a single list while seven domains were listed in two different lists. Moreover, when we checked all the effective second level domain names (e2LD) against the same lists, we only found 15 out of 17,502 (a minuscule 0.08%) e2LDs listed in one or more of the lists — with 11 e2TLDs being listed in a single list while four eTLDs were listed in two different lists. Table 13 depicts these results.

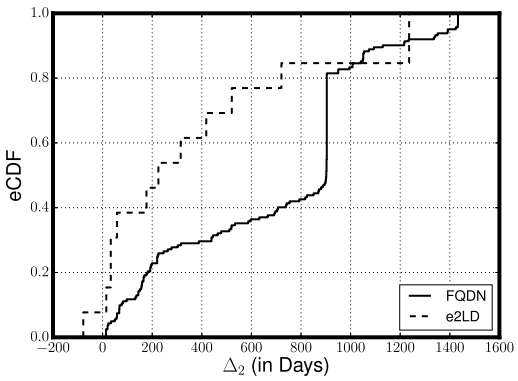
5.1.2 Potential benefits of sharing intelligence

Diving a bit deeper into the blacklisted domains, we wanted to measure the timeliness of the blacklist updates. To achieve this, we computed two metrics Δ_1 and Δ_2. For a blacklisted SMS-spam domain d, $\Delta_1(d)$ measures the difference in days between the earliest date the SMS-spam domain was seen on a blacklist and the earliest date the domain was seen in an SMS-spam message in our complaint repository. $\Delta_2(d)$ measures the difference in days between the earliest date the domain was seen in a blacklist and the earliest date it was looked up, according to the passive DNS visibility we obtained.

Figure 25(a) shows the empirical cumulative distribution (eCDF) of Δ_1 over all blacklisted domains. We show two plots, one for the FQDNs and the other for the e2LDs. A positive value for Δ_1 means that the blacklisting happened after the earliest complaint was received, whereas a negative value implies that the blacklisting happened before the earliest complaint was received. From the eCDF of FQDNs, it is clear that around 94% of the blacklisted FQDNs were blacklisted after the complaint was received ranging from zero to 1,393 days. It is clear that the blacklists are rather slow in incorporating these abusive domains. In some cases, about 6% FQDNs were



(a) eCDF of Δ_1



(b) eCDF of Δ_2

Figure 25: Timeliness of sharing intelligence: telephony to Internet.

blacklisted even before a complaint was received, indicating that sometimes either the SMS-spam is not reported on time or existing abuse domains related to traditional spam are being reused to cater to cross-channel spam. We observed a similar pattern in the case of e2LD.

Figure 25(b) shows the eCDF for Δ_2 for FQDNs and e2LDs. A positive value for Δ_2 means that the blacklisting happened after the earliest pDNS lookup as seen by our sensors, whereas a negative value implies that the blacklisting happened before the earliest pDNS lookup as seen in the pDNS database. In majority of the cases we observed a huge lag in the timeliness of the blacklist update. The lag ranged from 13 to 1433 days in the case of FQDNs and from -78 to 1506 days (only one negative

value was seen) in the case of e2LDs. Although these findings are for a relatively small number of domains (those that ever appeared in a blacklist), it is clear that the blacklists appear to be lagging in discovering SMS-spam domains.

5.2 Sharing of Intelligence: Internet to Telephony

Just like we showed that the telephony channel intelligence can benefit the Internet channel, the Internet channel can be a useful and untapped source of intelligence which can benefit the telephony channel. TSS idenfied via online search and ads and data collected around it on the Internet channel can be used to protect the telephony channel victims from cross-channel abuse. In this section, we elaborate on these benefits.

5.2.1 Internet abuse and its relation to Telephony Blacklists

To provide evidence of the lack of intelligence sharing from online datasets to identify abusive telephony end-points, we take phone numbers from an emerging set of public blacklists (PBL) [1, 32] that also offer telephony intelligence, and check/verify if any of the 3,365 phone numbers present in the search and ad based TSS webpages detected by the X-TSS system, also appear in these lists. We collected data from these lists beginning Jan 2014 up until April 2017, encompassing the AD/SR data collection period, which allows us to make fair comparisons.

We start with 800notes.com, which is a crowdsourced directory of unknown callers. It consists of complaints by users who post about telephony scams, not just technical support scams. We extracted phone numbers appearing in the complaints and compared them with those identified by us via online search and ad based TSS websites. We found that among phone numbers found on TSS websites identified by our system, only 16.8% toll-free numbers were also present in the complaint reports. This indicates the potential for improvement in the crowdsourced method of gathering telephony abuse intelligence. One way to do it would be to augment the complaints

Table 14: Coverage in Internet to telephony intelligence sharing scenario.

Blacklist Name	Phone Number
Malwarebytes TSS List	20.3%
800notes.com	16.8%
Cumulative	26.1%

based intelligence with automated intelligence generated by systems such as the X-TSS system presented in Chapter 4.

Next, we look at a more exclusive TSS blacklist released by Malwarebytes [32]. These public blacklists are specific to TSS and are regularly updated. According to their website, they use both crowdsourced and internal investigations to generate the list. Over time there were a total of 1,705 phone numbers listed on the Malwarebytes list. On comparing the two sets, we found that only 20.3% from our TSS dataset were also present in the telephony blacklist provided by Malwarebytes. This finding reinforces the previous one in suggesting that phone numbers used in cross-channel abuse are going largely undetected by existing telephony channel blacklists.

We also check popular call-blocking applications, such as TrueCaller and Mr. Number and find that less than 10% phone numbers from our lists are present in them. Miramirkhani el al. [80] made similar observations in their study on technical support scams. We suspect strongly that this low coverage is due to two reasons. First, the intelligence from the online channel is not getting integrated into these mobile application blocklists, like in the case of 800notes.com and Malwarebytes. Second, these call-blocking mobile applications are geared more towards blocking inbound phone communication (IPC), such as unverified robocalls, rather than blocking outbound phone communication (OPC) which is what happens when the victim is coerced into calling a TSS phone numbers when he/she lands on the TSS webpage associated with online search-and-ad abuse.

Even cumulatively, only 26.1% phone numbers identified by our system were

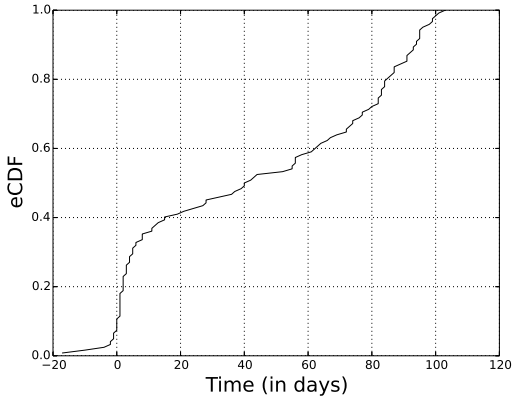


Figure 26: Timeliness of sharing intelligence: Internet to telephony.

present across all blacklists that included phone numbers. This suggests that while the notion of creating exclusive phone blacklists is a novel idea, cross-channel intelligence sharing can further augment these lists to include phone numbers currently going undetected due to a lack of intelligence flow from the online channel. Table 14 summarized these results.

5.2.2 Potential benefits of sharing intelligence

Once a victim calls the phone number listed on the fake TSS website, a call center operator uses voice-based interactions to social engineer the victim to pay up for the fake/unwanted technical support services.

Like in Section 5.1.2, we first dive a bit deeper into the blacklisted phone numbers, to measure the average time difference between when our X-TSS system first detected the abusive phone number and the earliest date on which the phone number was listed on any of the telephony blacklists mentioned previously. Figure 26 shows the spread of the timeliness factor for all phone numbers detected by both our system and the telephony blacklists. A positive value on the x-axis indicates that our system was able to detect the phone number before any of the blacklists while a negative value indicates otherwise. As we can see, our system does a better job 90% of the time, as compared to the aforementioned telephony blacklists, in marking a phone number

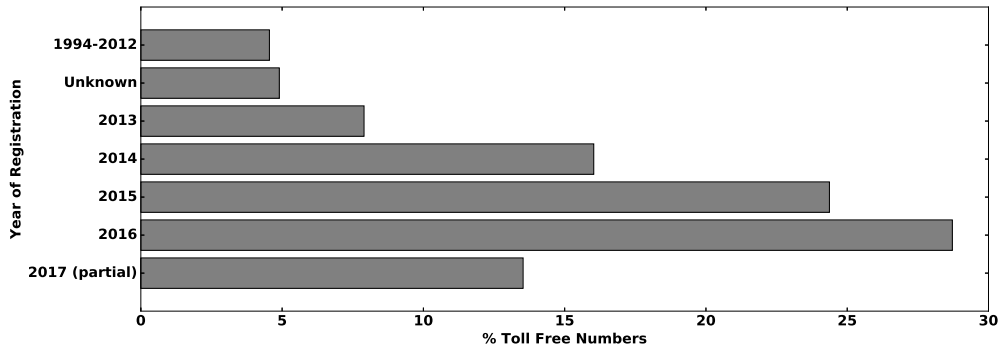


Figure 27: Distribution based on the year of registration of TSS phone numbers.

associated with technical support scams. On average our system was able to detect the blacklisted phone number ∼42 days in advance of it appearing on any blacklist. While it is difficult to measure the exact number of victims that could have been saved from a scam relying on any of these numbers, it is nevertheless clear that tapping the intelligence from the online channel would have bolstered the telephony channel defenses.

Telephony Abuse Attribution: Next, based on data from *tollfreenumbers.com*, we conduct an analysis of the toll-free numbers listed on the TSS webpages to gain insights into the telephony infrastructure that would have largely gone undetected if the intelligence from the online channel was missing. To do this, we look at two attributes associated with each toll-free number, while it was being abused: (i) the age of the toll-free number, and (ii) the toll-free number provider. We conduct this analysis for 3,365 unique toll-free numbers found on technical support scam webpages.

Registration Date/Age and Providers: The age of a toll-free number gives us an idea of when the number was purchased and registered by the technical support scammer who abuses it. We estimate this by fetching data that tells us the year in which the number last changed ownership and whether it is in *active use*. With both these factors combined, we can estimate the earliest possible time (not the exact time) when the organization or individual responsible for the account to which the toll-free

Table 15: Top 20 most abused toll free number providers.

Provider	% toll free numbers
WilTel Communications 866-WILTEL-1	26.10
ATL (a third party resporg service) 800-508-5200	6.93
Bandwidth.com customer	6.78
Not Present in main list	5.20
Not allowed to mention or warn people about	4.94
AT&T	4.53
Verizon 800-483-3722	4.38
Five 9 Inc. (BPS01)(925-201-2207)	3.36
Twilio Inc (TWI01)(877-889-4546)	3.21
Customer of CenturyLink (LGT01)(800-860-1020)	2.80
Teleglobe International Corp 800-567-1950	1.88
RespOrg Solution (private resporg service) 904-318-7271	1.58
Callture.	1.42
Dow Networks now AVOXI 770-937-9735	1.37
TimeShift (aka Onebox) 323-817-3220	1.22
CallSource 888-763-6200 or 800-500-4433	1.07
360 Networks USA.	1.07
Cimco Communications	0.91
McLeod USA (merged with Paetec) 800-593-1177	0.86
RingRevenue Inc. (YVR01)(805-617-1175)	0.86
Others	19.53
Total	100.0

number is linked could have potentially begun the abuse. Figure 27 plots the relative percentage of active toll-free numbers based on the year in which it last changed ownership. Close to 16% toll free numbers were registered in 2014, 24.4% in 2015, 28.7% in 2016 and 13.5% in early 2017, totaling to 82.6% of all toll free numbers in the period between Jan 2014 - Mar 2017. We were unable to find any information for 4.9%, and the remaining 12.5% were registered prior to 2014. The relatively recent timing of these registrations and their volume suggests that search-and-ad abuse TSS scams are a more recent phenomenon and are on the rise.

Even though the biggest provider of TSS-related toll-free phone numbers is WilTel Communications, contrary to the findings of Miramirkhani et al. [80], who find that four providers account for more than 90% of the phone numbers, we observe a much larger pool of providers, each responsible for a smaller fraction of numbers. The top four providers account for less than 40% of the identified TSS phone numbers.

Table 15 shows the relative percentage of toll-free numbers as a function of the

top 20 providers to which they belong. We notice that a single provider WilTel Communications contributes to nearly 26% of the toll-free numbers found in outbound phone communication (OPC) based fake technical support scam webpages. The other heavily abused providers include ATL Communication, and Bandwidth.com. Twilio Inc, a cloud communications platform as a service (PaaS) provider and RingRevenue were listed in the top four most abuse providers by Miramirkhani et al. [80] but we observed less abuse of these providers amongst the search-and-ad abuse based TSS.

More intriguing were numbers that "were not present in the main list of providers" and ones that were labeled as "not allowed to mention or warn people about" which we list together in the "Unable to verify" category. These comprise about 5% each (totaling to 10.14%) of the toll-free numbers seen. It is hard to tell whether providers are allowing this abuse knowingly or unknowingly as the toll-free numbers are paid for by the account holder and there is hardly any incentive for the provider to take them down as it is a source of revenue for them.

To conclude, in this section, using cross-channel abuse intelligence on TSS originating from the Internet channel, we were able to show how it can potentially benefit the telephony channel by improving coverage, timeliness and attribution of phone numbers involved in these scams.

5.3 Disseminating Cross-Channel Abuse Intelligence

Systems such as `CHURN` and `X-TSS` are useful to generate cross-channel abuse intelligence which can be valuable as shown in this chapter. However, the intelligence generated by these systems, needs to translate into protecting the end user. Since, both telephony and online channels are complex infrastructures, we discuss realistic ways to disseminate cross-channel abuse intelligence within the existing setup. In the online channel, new intelligence about domains and IP infrastructure generated by systems such as ours can be provided to network operators and Internet Service

Providers (ISPs) who can integrate this intelligence into existing network monitoring and protection systems deployed by them. This can be done in a format similar to popularly utilized, domain and IP blacklists. For online search-and-ad based technical support scams, the intelligence on abusive domains can be provided to the search engine operators directly or disseminated independently via a centralized update mechanism to popular browser extensions such as Personal Blocklist [34] (by Google) which provide the functionality to warn users about suspicious ADs/SRs by either excluding them from the search results completely or intercepting the request to such webpages with a warning page. On the telephony channel, the intelligence on technical support phone numbers can be integrated into applications that warn the user not just about incoming nuisance calls but also outgoing calls to toll-free numbers supporting technical support scams. Intelligence on SMS-spam domains can be passed onto companies such as Cloudmark [11] that inspect the payload in the cloud before transmitting the SMS message. In essence, cross-channel abuse intelligence can be integrated into protection mechanisms at both the end-point level such as in Internet browsers and mobile applications as well as at the network level either at its edge (eg. routers) or in the network elements that provide functionality to secure the network from undesired traffic.

5.4 Summary

We provide evidence that supports the claim that sharing of intellingence across the Internet and telephony channels is not occuring and that such sharing could be beneficial. In Section 5.1.1 we provided clear evidence that traditional domain and IP reputation feeds are failing to include the cross-channel SMS-spam domains even in a postmortem way and demonstrated that telephony datasets can help generate intelligence about previously unknown Internet abuse infrastructure. Further in Section 5.1.2 we showed, in effect, that if these telephony abuse datasets were tapped

during the period of abuse, the spurious domains hosted on the Internet could have been taken down much earlier.

We also explained intelligence sharing from the online to the telephony channel. In Sections 5.2.1, we showed that 73.9% phone numbers identified by our system, as associated with abusive online TSS infrastructure did not appear on phone blacklists or in telephony complaint datasets. This shows that if these online abuse datasets were tapped during the period of abuse, the abusive and previously unidentified phone numbers on the telephony channel could have been unearthed. In Section 5.2.2, we showed, in effect, that if these online abuse datasets on TSS were tapped during the period of abuse, the abuse phone numbers on the telephony channel could have been identified and flagged much earlier. Further, we are able to analyze the characteristics associated with these abusive and previously undetected phone numbers, such as the age and provider of the phone number, which helped us gain further insights into the underground telephony infrastructure supporting cross-channel abuse. This would not have been possible without tapping into telephony intelligence existing in the online channel. Lastly, we discussed ways in which cross-channel abuse intelligence can be disseminated to protect end users.

CHAPTER VI

CONCLUSION AND FUTURE WORK

6.1 Dissertation Summary and Contributions

In this dissertation, we show that cross-channel abuse poses a significant threat to consumers and propose methods to understand it. Using large scale threat intelligence data collected from different vantage points in the converged communications infrastructure landscape, we expose and better understand two cross-channel abuse cases: SMS-based scams and technical support scams (TSS). We then leverage this understanding to motivate the need for cross-channel intelligence sharing that could bolster existing defense mechanisms such as blacklists/blocklists on both the channels. Some of the contributions made by this dissertation include:

- We introduce the notion of cross-channel attacks, that leverage both telephony and Internet resources to victimize users, and place this new class of abuse in the context of traditionally defined Internet and telephony abuse.

- By using a data driven approach, that leverages several sources of abuse and ground truth data across the telephony and Internet channels, including crowd-sourced telephony intelligence, web threat intelligence on the abusive domain names, web templates and abusive IPs, we measure, analyze and understand two distinct cross-channel abuse cases: i) text-messaging abuse, and ii) technical support scams (TSS).

- To understand cross-channel text-messaging abuse, we design a cross-channel attribution system, `CHURN`, to automate the collection and analysis of SMS-spam abuse containing URLs. The proposed system is able to collect data about large

SMS abuse campaigns and analyze their passive DNS records and supporting website properties. It uses a hierarchical clustering technique that employs network level, application level, and popularity-based statistical features to cluster related SMS-spam domain names into campaigns over time. Using the system we are able to observe and measure the extent and effectiveness of cross-channel SMS-spam and reveal properties associated with the underlying infrastucture supporting the scam.

- To understand TSS from a cross-channel perspective, we develop and deploy a system `X-TSS`, that collects detailed information about online search and advertisement tactics used in the latest TSS and then analyzes the underlying infrastructure behind these scams. Using a known corpus of TSS webpages, the system systematically construct queries and uses multiple search engines to find TSS resources such as URIs, redirection chains, domains, and webpages that can be reached either from organic links returned by search engines or advertisements displayed by them. The system also uses network level amplification techniques to discover additional TSS infrastructure that may be difficult to identify via search results and advertisements only. Using the collected data, the system is able to reveal the tactics and infrastructure behind these evolving and sophisticated scams.

- Finally, we show how existing defenses like blacklists on the Internet and telephony channels are insufficient when dealing with cross-channel attacks. We provide evidence for sharing intelligence generated from systems such as `CHURN` and `X-TSS`, be shared to create a cross-channel security platform that enhances defenses across the two channels from an operational perspective while at the same time increasing the situational awareness around cross-channel abuse.

6.2 Limitations

We take a data-driven approach in this dissertation which suffers from certain limitations. Data collected and analyzed by CHURN, which includes consumer complaints and passive DNS data, is primarily US-centric, making it difficult to generalize the findings to other parts of the world. Indeed, cross-channel spam trends could be different in Europe or Asia as compared to the US. However, our attribution system, CHURN is designed to be easily deployable elsewhere, without much change.

CHURN's evaluation is based on a limited set of labeled data/ground truth. Although, we consciouly made an effort to label data that is representative of all the spam domains under consideration, by randomizing the selection process for manually inspecting the domains, we recognize the need to scale this experiment and plan to do it in the future while adding more capabilities to our system.

Like all real-world systems, X-TSS is also not without its limitations. Our choice of using PhantomJS for crawling search results and ads can, in principle, be detected by scammers who can use this knowledge to evade our monitors. We argue that replacing PhantomJS with a real browser is a relatively straightforward task which merely requires more hardware resources. Similarly, our choice of keeping our crawler stateless could lead to evasions which would again be avoided if one used a real, Selenium-driven, browser. Finally, while we provide clustering information for the discovered TSS, in the absence of ground truth, we are unable to attribute these clusters back to specific threat actors.

6.3 Future Work

With the research presented in this dissertation, we have shown that cross-channel abuse in converged communications infrastructure is a real threat which requires improved situational awareness by network operators, consumers and researchers. However, there are both direct extensions of this dissertation and broader research

problems that remain to be explored.

Among direct extensions of this work, there remain certain pieces to both `CHURN` and `X-TSS` that could be developed in the future. `CHURN` could be deployed in diverse geographic locations outside the US to get a global picture of cross-channel SMS abuse. Similarly, the `X-TSS` system could be scaled to add more nodes to it that are located in diverse geographic locations. This would help us collect more data on TSS and observe glocal trends of these scams. Next, we try to outline some of the promising research directions that are related to this dissertation.

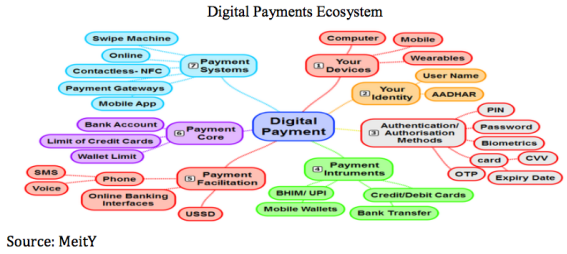


Figure 28: Envisioned digital payments ecosystem in India. Source: MeitY

Abuse in Converged Digital Payment/Currency Platforms Digital payment/currency platforms are increasingly getting integrated with the converged communications infrastructure. New architectures emerging in different parts of the world, such as the one shown in Figure 28, could give rise to cross-channel abuse that severely effects this sensitive sector. Research around preventing abuse in such scenarios would be extremely valuable.

Network Attribution Attributing cross-channel attacks to specific threat actors remains an unsolved problem. It is also a difficult problem since the converged communications network is complex with decentralized authority, that controls different portions of the network. This problem gets harder across geographic boundaries. Fine-grained data generated at different vantage points of the network could be the key in conducting timely and accurate attribution. Research in this area would also be extremenly valuable to the community.

REFERENCES

[1] "800notes - Directory of UNKNOWN Callers." http://800notes.com/.

[2] "abuse.ch - the swiss security blog." https://www.abuse.ch/.

[3] "Active DNS Project." https://www.activednsproject.org/.

[4] "Alexa Top Sites." http://www.alexa.com/topsites.

[5] "Alexa Topsites." http://www.alexa.com/topsites.

[6] "BeautifulSoup." https://pypi.python.org/pypi/beautifulsoup4.

[7] "Bing Ads bans ads from third-party tech support services." https://searchengineland.com/bing-bans-third-party-tech-support-ads-249356.

[8] "Bing brings in blanket ban on online tech support ads." https://nakedsecurity.sophos.com/2016/05/13/bing-brings-in-blanket-ban-on-online-tech-support-ads/.

[9] "Bing Search API." http://datamarket.azure.com/dataset/bing/search.

[10] "Block search indexing with 'noindex'." https://support.google.com/webmasters/answer/93710?hl=en.

[11] "CloudMark Intelligent Network Security." https://www.cloudmark.com/en.

[12] "Exposing Cross-Channel Abuse in Converged Communications Infrastructure with Text-Messaging Scams." https://smartech.gatech.edu/handle/1853/56635.

[13] "Federal Trade Commission FTC Complaint Assistant." https://www.ftccomplaintassistant.gov/crnt&panel1-1.

[14] "FTC - Tech Support Scams." https://www.consumer.ftc.gov/articles/0346-tech-support-scams.

[15] "FTC on Financial Freedom." https://www.ftc.gov/enforcement/cases-proceedings/092-3056/financial-freedom-processing-inc-formerly-known-financial.

[16] "FTC on Payday Lending." https://www.ftc.gov/news-events/media-resources/consumer-finance/payday-lending.

[17] "FTC Robocall Challenge." https://robocall.devpost.com/.

[18] "Geek Squad Services - Best Buy." http://www.bestbuy.com/site/electronics/geek-squad/pcmcat138100050018.c?id=pcmcat138100050018.

[19] "Goentry.com - how to remove?." http://www.2-remove-virus.com/nl/goentry-com-hoe-te-verwijderen/.

[20] "Google Custom Search." https://developers.google.com/custom-search/docs/overview.

[21] "Google Safe Browsing." https://www.google.com/transparencyreport/safebrowsing/.

[22] "hphosts." http://www.hosts-file.net/.

[23] "Indian police arrest alleged ringleader of IRS scam." http://money.cnn.com/2017/04/09/news/tax-scam-india-arrest-ringleader/.

[24] "India's Call-Center Talents Put to a Criminal Use: Swindling Americans." https://www.nytimes.com/2017/01/03/world/asia/india-call-centers-fraud-americans.html?_r=0.

[25] "Internet archive: Wayback machine." https://archive.org/web/.

[26] "I.T. Mate Product Support." http://support.it-mate.co.uk/.

[27] "Keyword Planner." https://adwords.google.com/KeywordPlanner.

[28] "Macrofix Wiki." http://tech-support-scam.wikia.com/wiki/Macrofix.

[29] "Malc0de database.." http://malc0de.com/database/.

[30] "Malc0de database." http://malc0de.com/database/.

[31] "Malware Domain List." http://www.malwaredomainlist.com/.

[32] "Malwarebytes Lab." https://blog.malwarebytes.com/tech-support-scams/.

[33] "N-Grams." https://lagunita.stanford.edu/c4x/Engineering/CS-224N/asset/slp4.pdf.

[34] "Personal Blocklist (by Google) - Chrome Web Store." https://chrome.google.com/webstore/detail/personal-blocklist-by-goo/nolijncfnkgaikbjbdaogikpmpbdcdef?hl=en.

[35] "PhantomJS." http://phantomjs.org/.

[36] "Python language bindings for Selenium WebDriver." https://pypi.python.org/pypi/selenium.

[37] "Remove Goentry.com (FREE GUIDE)." https://www.zemana.com/en-US/removal-guide/remove-goentry.com.

[38] "sagadc summary." http://dns-bh.sagadc.org/.

[39] "Searching For 'Facebook Customer Service' Can Lead To A Scam." http://www.npr.org/sections/alltechconsidered/2017/01/31/511824829/-facebook-customer-service-is-a-scam-literally.

[40] "SMS Phishers Exploit Twilio and ow.ly to Steal Mobile Account Logins." http://blog.cloudmark.com/2014/02/13/sms-phishers-exploit-twilio-and-owly-to-steal-mobile-account-logins/.

[41] "SMSWatchDog." http://www.smswatchdog.com.

[42] "SPAMHaus Blocklist." https://www.spamhaus.org/lookup/.

[43] "Suspicious domains - sans internet storm center." https://isc.sans.edu/suspicious_domains.html.

[44] "Tech support scams persist with increasingly crafty techniques.." https://blogs.technet.microsoft.com/mmpc/2017/04/03/tech-support-scams-persist-with-increasingly-crafty-techniques/.

[45] "Text Spammers Settle FTC Charges They Illegally Sent Consumers Bogus Offers for 'Free' Gift Cards." https://www.ftc.gov/news-events/press-releases/2013/09/text-spammers-settle-ftc-charges-they-illegally-sent-consumers.

[46] "The ICE Cyber Crime Center Virus Removal Guide." https://malwaretips.com/blogs/ice-cyber-crime-center-removal/.

[47] "Tropo." https://www.tropo.com.

[48] "Twilio." http://www.twilio.com.

[49] "Universal Event Tracking." http://help.bingads.microsoft.com/apex/index/3/en-us/n5012.

[50] "VirusTotal." https://www.virustotal.com/.

[51] "What kind of SMS messages are not allowed to be sent using Twilio?." https://www.twilio.com/help/faq/sms/what-kind-of-sms-messages-are-not-allowed-to-be-sent-using-twilio.

[52] "Your very own SMS Internet gateway with Arduino." http://x-ian.net/2012/10/09/your-very-own-sms-internet-gateway-with-arduino/.

[53] Anderson, D. S., Fleizach, C., Savage, S., and Voelker, G. M., "Spamscatter: Characterizing internet scam hosting infrastructure," in *Proceedings of the 16th USENIX Security Symposium, Boston, MA, USA, August 6-10, 2007* (Provos, N., ed.), USENIX Association, 2007.

[54] Antonakakis, M., Perdisci, R., Dagon, D., Lee, W., and Feamster, N., "Building a dynamic reputation system for DNS," in *19th USENIX Security Symposium, Washington, DC, USA, August 11-13, 2010, Proceedings*, pp. 273–290, USENIX Association, 2010.

[55] Antonakakis, M., Perdisci, R., Lee, W., Vasiloglou, N., and Dagon, D., "Detecting malware domains in the upper DNS hierarchy," in *the Proceedings of 20th USENIX Security Symposium (USENIX Security '11)*, 2011.

[56] Antonakakis, M., Perdisci, R., Nadji, Y., Vasiloglou, N., Abu-Nimeh, S., Lee, W., and Dagon, D., "From Throw-Away Traffic to Bots: Detecting the Rise of DGA-Based Malware," in *the Proceedings of 21th USENIX Security Symposium (USENIX Security '12)*, 2012.

[57] Antonakakis, M., Perdisci, R., Nadji, Y., Vasiloglou, N., Abu-Nimeh, S., Lee, W., and Dagon, D., "From throw-away traffic to bots: Detecting the rise of dga-based malware," in *Presented as part of the 21st USENIX Security Symposium (USENIX Security 12)*, (Bellevue, WA), pp. 491–506, USENIX, 2012.

[58] Balasubramaniyan, V. A., Poonawalla, A., Ahamad, M., Hunter, M. T., and Traynor, P., "Pindr0p: Using single-ended audio features to determine call provenance," in *Proceedings of the 17th ACM Conference on Computer and Communications Security*, CCS '10, (New York, NY, USA), pp. 109–120, ACM, 2010.

[59] Bilge, L., Kirda, E., Kruegel, C., and Balduzzi, M., "EXPOSURE: finding malicious domains using passive DNS analysis," in *Proceedings of the Network and Distributed System Security Symposium, NDSS 2011, San Diego, California, USA, 6th February - 9th February 2011*, The Internet Society, 2011.

[60] Boggs, N., Wang, W., Mathur, S., Coskun, B., and Pincock, C., "Discovery of emergent malicious campaigns in cellular networks," in *Proceedings of the 29th Annual Computer Security Applications Conference*, ACSAC '13, (New York, NY, USA), pp. 29–38, ACM, 2013.

[61] Burges, C. J., "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[62] Chen, Y., Nadji, Y., Gómez, R. R., Antonakakis, M., and Dagon, D., "Measuring network reputation in the ad-bidding process," in *Detection of Intrusions and Malware, and Vulnerability Assessment - 14th International Conference, DIMVA 2017, Bonn, Germany, July 6-7, 2017, Proceedings*, pp. 388–409, 2017.

[63] Dave, V., Guha, S., and Zhang, Y., "Viceroi: catching click-spam in search ad networks," CCS 2013.

[64] Forbes, "Hackers Are Hijacking Phone Numbers And Breaking Into Email, Bank Accounts: How To Protect Yourself.." `http://bit.ly/2rauxNY`, 2017.

[65] Forney, G.D., J., "The viterbi algorithm," *Proceedings of the IEEE*, vol. 61, pp. 268–278, March 1973.

[66] Garera, S., Provos, N., Chew, M., and Rubin, A. D., "A framework for detection and measurement of phishing attacks," in *Proceedings of the 2007 ACM Workshop on Recurring Malcode*, WORM '07, (New York, NY, USA), pp. 1–8, ACM, 2007.

[67] Grier, C., Thomas, K., Paxson, V., and Zhang, M., "@spam: The underground on 140 characters or less," in *Proceedings of the 17th ACM Conference on Computer and Communications Security*, CCS '10, (New York, NY, USA), pp. 27–37, ACM, 2010.

[68] Gupta, P., Srinivasan, B., Balasubramaniyan, V., and Ahamad, M., "Phoneypot: Data-driven understanding of telephony threats," in *22nd Annual Network and Distributed System Security Symposium, NDSS 2015, San Diego, California, USA, February 8-11, 2015*, The Internet Society, 2015.

[69] Hao, S., Syed, N. A., Feamster, N., Gray, A. G., and Krasser, S., "Detecting spammers with snare: Spatio-temporal network-level automatic reputation engine," in *Proceedings of the 18th Conference on USENIX Security Symposium*, SSYM'09, (Berkeley, CA, USA), pp. 101–118, USENIX Association, 2009.

[70] Hindustan Times, "Scare and sell: Here's how an Indian call centre cheated foreign computer owners." `http://bit.ly/2oj2Rpz`, 2017.

[71] Jiang, N., Jin, Y., Skudlark, A., and Zhang, Z.-L., "Greystar: Fast and accurate detection of sms spam numbers in large cellular networks using grey phone space," in *Proceedings of the 22Nd USENIX Conference on Security*, SEC'13, (Berkeley, CA, USA), pp. 1–16, USENIX Association, 2013.

[72] Kapoor, S., Sharma, S., and Srinivasan, B., "Clustering devices in an internet of things ('iot')," Mar. 11 2014. US Patent 8,671,099.

[73] Kapoor, S., Sharma, S., and Srinivasan, B., "Attribute-based identification schemes for objects in internet of things," July 23 2013. US Patent 8,495,072.

[74] Leontiadis, N., Moore, T., and Christin, N., "Measuring and analyzing search-redirection attacks in the illicit online prescription drug trade," USENIX Security 2011.

[75] Levchenko, K., Pitsillidis, A., Chachra, N., Enright, B., Félegyházi, M., Grier, C., Halvorson, T., Kanich, C., Kreibich, C., Liu, H., and others, "Click trajectories: End-to-end analysis of the spam value chain," in *Security and Privacy (SP), 2011 IEEE Symposium on*, pp. 431–446, IEEE, 2011.

[76] Lever, C., Antonakakis, M., Reaves, B., Traynor, P., and Lee, W., "The core of the matter: Analyzing malicious traffic in cellular carriers," in *20th Annual Network and Distributed System Security Symposium, NDSS 2013, San Diego, California, USA, February 24-27, 2013*, The Internet Society, 2013.

[77] Maggi, F., "Are the con artists back? A preliminary analysis of modern phone frauds," in *10th IEEE International Conference on Computer and Information Technology, CIT 2010, Bradford, West Yorkshire, UK, June 29-July 1, 2010*, pp. 824–831, 2010.

[78] Manning, C. D. and Schütze, H., *Foundations of statistical natural language processing*, vol. 999. MIT Press, 1999.

[79] Ministry of Finance, Government of India, "Report of the working group for setting up of computer emergency response team in the financial sector (CERT-Fin)." `http://dea.gov.in/sites/default/files/Press-CERT-FinReport.pdf`, 2017.

[80] Miramirkhani, N., Starov, O., and Nikiforakis, N., "Dial one for scam: A large-scale analysis of technical support scams," NDSS 2017.

[81] Murynets, I. and Jover, R. P., "Crime scene investigation: SMS spam data analysis," in *Proceedings of the 12th ACM SIGCOMM Conference on Internet Measurement, IMC '12, Boston, MA, USA, November 14-16, 2012* (Byers, J. W., Kurose, J., Mahajan, R., and Snoeren, A. C., eds.), pp. 441–452, ACM, 2012.

[82] Nazario, J. and Holz, T., "As the net churns: Fast-flux botnet observations," in *3rd International Conference on Malicious and Unwanted Software, MALWARE 2008, Alexandria, Virginia, USA, October 7-8, 2008*, pp. 24–31, 2008.

[83] Pelleg, D., Moore, A. W., and others, "X-means: Extending k-means with efficient estimation of the number of clusters.," in *ICML*, pp. 727–734, 2000.

[84] Polakis, I., Petsas, T., Markatos, E. P., and Antonatos, S., "A systematic characterization of IM threats using honeypots," in *Proceedings of the Network and Distributed System Security Symposium, NDSS 2010, San Diego, California, USA, 28th February - 3rd March 2010*, 2010.

[85] Positive Technologies, "Vulnerabilities in mobile networks opens bitcoin wallets to hackers.." https://www.ptsecurity.com/ww-en/about/news/285038/, 2017.

[86] Sahin, M., Relieu, M., and Francillon, A., "Using chatbots against voice spam: Analyzing lennyâĂŹs effectiveness," SOUPS 2017.

[87] Salton, G. and McGill, M. J., *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, Inc., 1986.

[88] searchengineland.com, "Tech support scams remain at the top of the list of bad actors that search engines have to keep fighting.." https://selnd.com/24jskRr, 2016.

[89] Sengupta, D., Goswami, A., Schwan, K., and Pallavi, K., "Scheduling multi-tenant cloud workloads on accelerator-based systems," in *International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2014, New Orleans, LA, USA, November 16-21, 2014*, pp. 513–524, 2014.

[90] Sengupta, D., Song, S. L., Agarwal, K., and Schwan, K., "Graphreduce: processing large-scale graphs on accelerator-based systems," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2015, Austin, TX, USA, November 15-20, 2015*, pp. 28:1–28:12, 2015.

[91] Sengupta, D., Sundaram, N., Zhu, X., Willke, T. L., Young, J., Wolf, M., and Schwan, K., "Graphin: An online high performance incremental graph processing framework," in *Euro-Par 2016: Parallel Processing - 22nd International Conference on Parallel and Distributed Computing, Grenoble, France, August 24-26, 2016, Proceedings*, pp. 319–333, 2016.

[92] Sharma, S., Kapoor, S., Srinivasan, B. R., and Narula, M. S., "Hicho: Attributes based classification of ubiquitous devices," in *Mobile and Ubiquitous Systems: Computing, Networking, and Services - 8th International ICST Conference, MobiQuitous 2011, Copenhagen, Denmark, December 6-9, 2011, Revised Selected Papers* (Puiatti, A. and Gu, T., eds.), vol. 104 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 113–125, Springer, 2011.

[93] Srinivasan, B., Gupta, P., Antonakakis, M., and Ahamad, M., "Understanding cross-channel abuse with sms-spam support infrastructure attribution," in *Computer Security - ESORICS 2016 - 21st European Symposium on Research in Computer Security, Heraklion, Greece, September 26-30, 2016, Proceedings, Part I* (Askoxylakis, I. G., Ioannidis, S., Katsikas, S. K., and Meadows, C. A., eds.), vol. 9878 of *Lecture Notes in Computer Science*, pp. 3–26, Springer, 2016.

[94] Srinivasan, B., Kountouras, A., Miramirkhani, N., Alam, M., Nikiforakis, N., Antonakakis, M., and Ahamad, M., "By hook or by crook: Exposing the diverse abuse tactics of technical support scammers," *CoRR*, vol. abs/1709.08331, 2017.

[95] Stone-Gross, B., Cova, M., Cavallaro, L., Gilbert, B., Szydlowski, M., Kemmerer, R., Kruegel, C., and Vigna, G., "Your botnet is my botnet: Analysis of a botnet takeover," in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, CCS '09, (New York, NY, USA), pp. 635–647, ACM, 2009.

[96] Thomas, K., Grier, C., Ma, J., Paxson, V., and Song, D., "Design and evaluation of a real-time URL spam filtering service," in *32nd IEEE Symposium on Security and Privacy, S&P 2011, 22-25 May 2011, Berkeley, California, USA*, pp. 447–462, IEEE Computer Society, 2011.

[97] Thomas, K., Iatskiv, D., Bursztein, E., Pietraszek, T., Grier, C., and McCoy, D., "Dialing back abuse on phone verified accounts," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pp. 465–476, ACM, 2014.

[98] trustinads.org, "Bad Ads Trend Alert: Shining a Light on Tech Support Advertising Scams." `http://bit.ly/2ypcFpn`.

[99] Vissers, T., Joosen, W., and Nikiforakis, N., "Parking sensors: Analyzing and detecting parked domains," in *22nd Annual Network and Distributed System Security Symposium, NDSS 2015, San Diego, California, USA, February 8-11, 2014*, 2015.

[100] Wall, M. E., Rechtsteiner, A., and Rocha, L. M., "Singular value decomposition and principal component analysis," in *A practical approach to microarray data analysis*, pp. 91–109, Springer, 2003.

[101] Yadav, S., Reddy, A. K. K., Reddy, A., and Ranjan, S., "Detecting algorithmically generated malicious domain names," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pp. 48–61, ACM, 2010.

VITA

Bharat Srinivasan is a PhD candidate in Computer Science at the College of Computing at the Georgia Institute of Technology, USA. He is also affiliated with the Institute of Information Security and Privacy (IISP) at Georgia Tech. His research at Georgia Tech focuses on securing networks and critical communications infrastrucuture by designing automated systems that leverage machine learning techniques.